

THESIS / THÈSE

MASTER EN SCIENCES INFORMATIQUES

Contribution à la communication entre équipements de traitement de données Liaison entre équipements, échange interactif, transfert de fichiers

Valentin, Didier

Award date:
1986

Awarding institution:
Université de Namur

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Facultés Universitaires Notre-Dame de la Paix

Institut d'informatique

Contribution à la communication entre
équipements de traitement de données

Liaison entre équipements
Echange interactif
Transfert de fichiers

Mémoire présenté par

Didier VALENTIN

en vue de l'obtention

du titre de

Licencié et Maître en Informatique

Année académique 1985-1986

Nous remercions pour leur accueil la firme OFISA au sein de laquelle nous avons évolué pendant cinq mois et plus particulièrement Monsieur Desaulles dont les conseils nous furent précieux.

Nous tenons également à témoigner notre gratitude à notre directeur de mémoire, Monsieur Brunin.

AVANT PROPOS

L'objet du stage qui a précédé ce mémoire, était la communication entre ordinateurs, de même marque ou de marques différentes, éloignés ou non, dédiés à des services de télématique ou de gestion, reliés par un système de réseau ou par lignes point à point.

Les objectifs de l'établissement d'une liaison de données entre équipements de traitement de données, consistaient d'une part en l'échange de fichiers entre les diverses machines, d'autre part en un dialogue interactif de terminal à terminal ou de terminal à une base de données.

La figure A.1 représente la future configuration type envisagée pour les diverses machines de la firme qui nous a accueilli.

On trouvera à la figure A.2 l'explication des abréviations employées.

La figure A.3 représente quant à elle la configuration de la commutation de données que l'entreprise va acquérir. Il s'agit en fait d'un PABX (Private Automatic Branch eXchange), appelé aussi régie d'abonnés.

Afin de permettre et de faciliter la réalisation future de la configuration type, nous nous sommes intéressés à diverses questions posées lors du stage.

Les problèmes rencontrés étaient de divers ordres.

Le premier type de problèmes concernait la liaison entre deux machines.

Nous nous sommes retrouvés face à toute une série d'interfaces normalisées.

La relation entre les interfaces, leur utilité, les normes, les organismes de normalisation, ne nous sont pas apparus comme clairement définis.

Faire le point sur les principales interfaces utilisées actuellement est l'un des buts de ce mémoire.

Le second type de problèmes concernait les protocoles utilisés pour la connexion entre terminal de traitement de données.

Nous nous sommes intéressés aux protocoles standards de liaison de données utilisés habituellement.

Nous avons établis les différences principales entre ces protocoles.

Nous nous sommes intéressés à leur utilité, aux relations qu'ils définissent entre partenaires, à leur procédure ainsi qu'à leur performance, et aux techniques utilisées pour permettre la transmission d'informations de type quelconque.

Afin de permettre la programmation future des deux types d'applications désirées (échange de fichiers, dialogue

interactif), nous avons établi un modèle consistant en les couches session et présentation, couches 5 et 6 du modèle ISO, appelé aussi modèle à 7 couches.

Pour établir ce modèle, nous nous sommes basés sur le mémoire traitant des couches supérieures de l'ISO, écrit par messieurs Henrion et Waltzing, ainsi que sur les couches supérieures du télétext, que nous avons abordés de façon très critique.

Le lecteur trouvera donc dans ce mémoire :

- au chapitre 1 : une introduction générale au modèle ISO et à son fonctionnement
- au chapitre 2 : une liste des principales interfaces physiques
- au chapitre 3 : les deux grandes catégories de protocoles de liaison de données utilisés
- au chapitre 4 : le niveau session de notre modèle pour l'échange interactif et le transfert de fichiers
- au chapitre 5 : le niveau présentation de notre modèle pour les deux mêmes applications

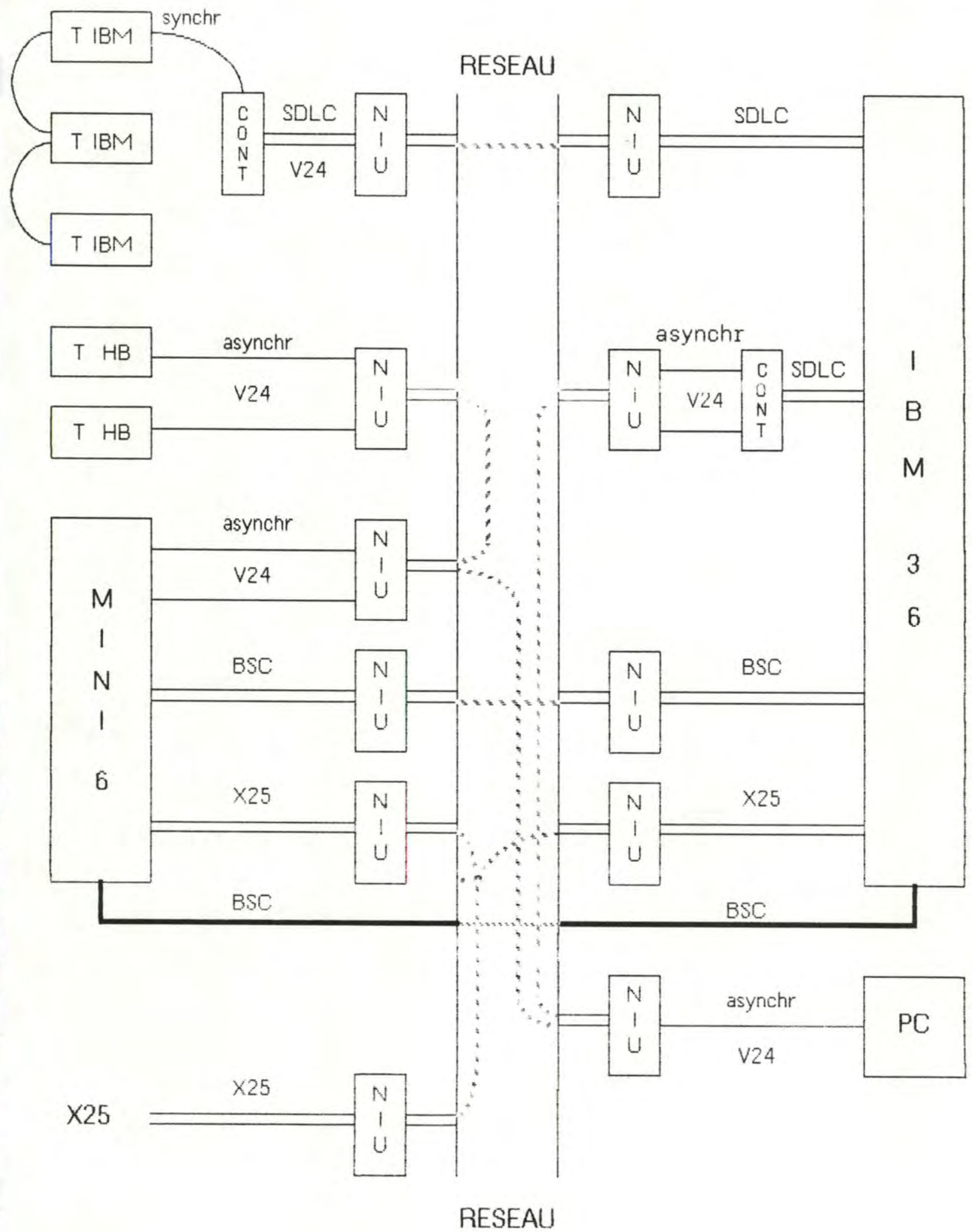
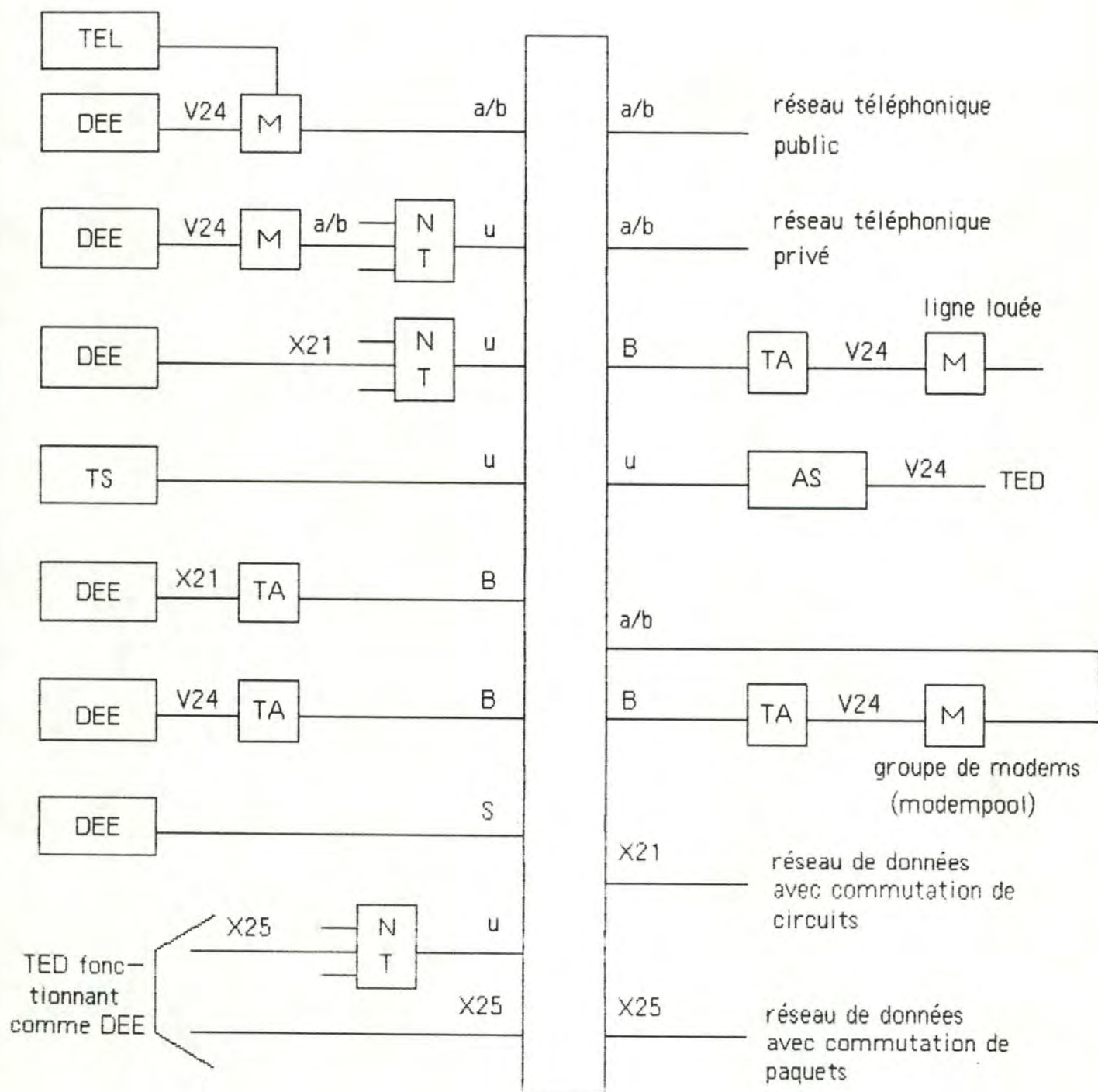


Figure A.1: Configuration type

| | |
|--------|---------------------------|
| T IBM | Terminal IBM |
| T HB | Terminal Honeywell Bull |
| MINI 6 | Ordinateur Honeywell Bull |
| IBM 36 | Ordinateur IBM |
| PC | Personal Computer |
| cont | contrôleur |
| NIU | Network Interface Unit |

Figure A.2: Abréviations de la figure A.1



M : modem

TA : adaptateur pour terminal

NT : terminaison de réseau

DEE : équipement de terminaison du circuit de données

AS : adaptateur spécialisé du PABX

TS : terminal spécialisé du PABX

TEL : téléphone

TED : traitement électronique de données

Figure A.3: Configuration de la commutation de données

TABLE DES MATIERES

| | | |
|---------|---|----|
| 0 | INTRODUCTION | 1 |
| 1 | MODELE OSI | 4 |
| 1.1 | Présentation générale | 5 |
| 1.1.1 | Découpe en couches | 5 |
| 1.1.1.1 | Introduction | 5 |
| 1.1.1.2 | Relations entre couches | 5 |
| 1.1.2 | Fonctionnement général | 6 |
| 1.1.2.1 | Implémentation du protocole vertical: le moniteur | 6 |
| 1.1.2.2 | Gestion du protocole horizontal: l'automate | 7 |
| 1.2 | Fonctions des différentes couches | 8 |
| 1.2.1 | Couche physique | 8 |
| 1.2.2 | Couche liaison de données | 8 |
| 1.2.3 | Couche acheminement (réseau) | 8 |
| 1.2.4 | Couche transport | 9 |
| 1.2.5 | Couche session | 9 |
| 1.2.6 | Couche présentation | 9 |
| 1.2.7 | Couche application | 10 |
| 2 | INTERFACES PHYSIQUES | 11 |
| 2.1 | Introduction | 12 |
| 2.1.1 | Définition de l'interface physique .. | 12 |
| 2.1.2 | Caractéristiques générales | 12 |
| 2.1.3 | Fonctions de l'interface physique ... | 13 |
| 2.1.4 | Organismes intéressés | 13 |
| 2.2 | Interfaces traditionnelles et pour réseau téléphonique | 14 |
| 2.2.1 | Interfaces de la série V, introduction | 14 |
| 2.2.2 | Recommandation V 24 | 15 |
| 2.2.2.1 | Vue générale des circuits et comparaison RS | 15 |
| 2.2.2.2 | Caractéristiques électriques de l'interface V24 | 22 |
| 2.2.2.3 | Caractéristiques mécaniques | 24 |

| | | |
|---------|--|----|
| 2.2.3 | Les modems | 25 |
| 2.2.3.1 | Introduction | 25 |
| 2.2.3.2 | Liste des modems et de leurs caractéristiques | 27 |
| 2.3 | Interfaces pour réseaux publics pour données | 28 |
| 2.3.1 | Introduction | 28 |
| 2.3.2 | La recommandation X 24 | 31 |
| 2.3.3 | Interface physique selon X 20 | 31 |
| 2.3.4 | Interface physique selon X 21 | 31 |
| 2.3.5 | Interfaces physiques pour X 25 | 35 |
| 2.3.5.1 | Mode synchrone | 35 |
| 2.3.5.2 | Mode asynchrone | 35 |
| 2.3.6 | Interfaces X 20 bis et X 21 bis | 36 |
| 2.4 | Interfaces pour le RNIS | 38 |
| 3 | PROTOCOLES DE LIAISON DE DONNEES | 39 |
| 3.1 | Introduction | 40 |
| 3.2 | Le protocole BSC | 41 |
| 3.2.1 | Introduction | 41 |
| 3.2.2 | Relation entre partenaires | 41 |
| 3.2.3 | Procédure 'stop and wait' | 42 |
| 3.2.4 | Réalisation de la transparence | 42 |
| 3.2.5 | Messages utilisés | 43 |
| 3.2.5.1 | Format des données | 43 |
| 3.2.5.2 | Séquence de supervision | 44 |
| 3.2.6 | Remarques générales | 45 |
| 3.3 | Les protocoles évolués | 46 |
| 3.3.1 | Introduction | 46 |
| 3.3.2 | Relation primaire secondaire | 46 |
| 3.3.3 | Procédure à fenêtre glissante | 47 |
| 3.3.4 | Réalisation de la transparence | 47 |
| 3.3.5 | Types de trames | 48 |
| 3.3.5.1 | Trame d'informations | 49 |
| 3.3.5.2 | Trames de supervision | 50 |
| 3.3.5.3 | Trames non numérotées | 51 |

| | | |
|---------|---|----|
| 4 | LA COUCHE SESSION | 52 |
| 4.1 | Introduction | 53 |
| 4.2 | La session d'échange de fichiers | 54 |
| 4.2.1 | Fonctions du protocole et procédure générale | 54 |
| 4.2.2 | Détail des fonctions et des procédures | 59 |
| 4.2.2.1 | Initialisation de l'échange | 59 |
| 4.2.2.2 | Echange | 60 |
| 4.2.2.3 | Contrôle de l'échange | 60 |
| 4.2.2.4 | Fin normale de session | 61 |
| 4.2.2.5 | Fin anormale de session | 61 |
| 4.2.3 | Gestion des erreurs au niveau 5 | 62 |
| 4.2.3.1 | Erreurs du niveau 5 | 62 |
| 4.2.3.2 | Erreurs au niveau 6 | 62 |
| 4.2.3.3 | Autres cas | 63 |
| 4.2.4 | Les timers | 64 |
| 4.3 | La session interactive | 65 |
| 4.3.1 | Introduction | 65 |
| 4.3.2 | Fonctions et mécanismes retenus | 66 |
| 4.3.2.1 | Le mécanisme du jeton | 66 |
| 4.3.2.2 | La suspension | 66 |
| 4.3.2.3 | Fin de session | 66 |
| 4.3.2.4 | Autres caractéristiques | 67 |
| 4.3.3 | Détail des fonctions et des procédures | 68 |
| 4.3.3.1 | Gestion du jeton | 68 |
| 4.3.3.2 | Gestion de la suspension | 68 |
| 4.3.3.3 | Procédure de fin de session | 72 |
| 4.3.3.4 | Initialisation | 73 |
| 4.3.3.5 | Echange interactif | 73 |
| 4.3.4 | Gestion des erreurs | 73 |
| 4.3.4.1 | Erreurs générales | 73 |
| 4.3.4.2 | Erreurs portant sur le jeton | 74 |
| 4.3.5 | Les timers | 75 |

| | | |
|---------|---|----|
| 5 | LA COUCHE PRESENTATION | 76 |
| 5.1 | Introduction | 77 |
| 5.2 | Echange de fichiers | 78 |
| 5.2.1 | Fonctions du protocole et procédure générale | 78 |
| 5.2.2 | Détail des fonctions et des procédures | 79 |
| 5.2.2.1 | Initialisation | 79 |
| 5.2.2.2 | Lancement et relancement de l'échange | 84 |
| 5.2.2.3 | Transfert | 85 |
| 5.2.2.4 | Fin normale de document | 87 |
| 5.2.2.5 | Fin anormale de document | 87 |
| 5.2.3 | Gestion des erreurs | 88 |
| 5.2.3.1 | Erreurs de timer | 88 |
| 5.2.3.2 | Autres erreurs de protocole | 88 |
| 5.2.4 | Les timers | 89 |
| 5.3 | Interactif | 90 |
| 5.3.1 | Introduction | 90 |
| 5.3.2 | Détail des fonctions et des primitives | 91 |
| 5.3.2.1 | Initialisation du niveau | 91 |
| 5.3.2.2 | Echange interactif | 94 |
| | Conclusion | 95 |
| | Bibliographie | 96 |

0 INTRODUCTION

La tendance actuelle des applications informatiques est à la répartition des systèmes, développés séparément et interconnectés par la suite.

L'envahissement du marché informatique par les mini-ordinateurs et les personal computers semble confirmer une évolution vers la décentralisation.

Les demandes d'interconnexion sont très diverses. Nous citerons à titre d'exemples : le raccordement de systèmes aux services de la télématique, la connexion de systèmes plus ou moins compatibles, ou non compatibles, la connexion de systèmes éloignés.

Ce phénomène a amené d'une part les constructeurs à développer des systèmes permettant aux ordinateurs de leur gamme de dialoguer, d'autre part aux grands organismes de normalisation d'imaginer des systèmes d'interconnexion hétérogènes pour l'ensemble des machines, ainsi que de standardiser les moyens de communication tels que les réseaux locaux ou les médias de communication publics nationaux et internationaux.

Parmi les organismes internationaux de normalisation, reprenons ici les plus importants :

CCITT : Comité Consultatif International Télégraphique
Téléphonique.

Ce département de l'UIT, dépendant de l'ONU,
travaille en commun avec l'ISO.

Il a développé notamment les avis V et X.

ISO : International Standard Organisation.

Cet organisme, dont le siège est à Genève,
a mis au point un modèle de référence
d'interconnexion de systèmes ouverts, connu
sous les abréviations RM/OSI (modèle à 7 couches).

Des associations se sont aussi formées sur les différents continents.

Nous trouvons entre autres :

- propres aux Etats Unis :

- EIA : Electronic Industries Association.
Norme RS.

- IEEE : Institute of Electrical and Electronics
Engineers.
Méthodes d'accès aux réseaux privés.

- en Europe :

- ECMA : European Computer Manufacturer Association.
Reprise de l'OSI.

- CEPT : Conférence Européenne des Postes
et Télécommunications.
Videotex,...

Le modèle RM/OSI, développé par l'ISO, a été suivi de façon plus ou moins fidèle par les diverses organisations et les grands constructeurs.
C'est le modèle que nous essaierons de suivre.

Nous ne reprendrons pas l'ensemble des sept couches, mais seulement celles qui sont concernées par le problème suivant :

- établir une ligne entre deux ordinateurs quelconques
- pouvoir converser et échanger des fichiers.

Nous nous intéresserons donc premièrement aux problèmes survenant lors de l'établissement d'une liaison de données. Dans le chapitre intitulé interfaces physiques, une liste non-exhaustive des interfaces physiques les plus connus et usités pour l'établissement d'un circuit de données sera dressée.

Dans le chapitre suivant, les protocoles de liaison de données, nous établirons un inventaire des principaux moyens existants, de leurs caractéristiques et de leurs applications pour la réalisation d'une liaison de données.

Ensuite, nous étudierons les niveaux supérieurs de l'OSI, niveaux 5 et 6, pour l'établissement d'une session interactive et de transfert de fichiers, avec leurs niveaux présentation correspondants.

1 MODELE OSI

1.1 Présentation générale

1.1.1 Découpe en couches

1.1.1.1 Introduction

Les systèmes informatiques sont des systèmes complexes qui nécessitent une organisation en fonctions clairement définies et séparées dans des niveaux.

Ce type d'organisation assure :

- l'indépendance de l'activité des niveaux entre eux
- la possibilité de cacher de l'information à chaque niveau
- le séquençement des événements dans le temps de niveau à niveau

La raison première, propre à la téléinformatique, est en fait de tamponner l'espace applications-réseaux, de telle sorte que la couche application, quelle qu'elle soit, reste toujours indépendante de celle réseau et vice-versa.

Le nombre de couches a été arrêté à sept.

Elles se nomment : niveau 1 physique, niveau 2 logique ou liaison de données, niveau 3 réseau ou acheminement, pour les couches de réseaux.

En ce qui concerne les couches d'application ou couches supérieures : niveau 4 transport, niveau 5 session, niveau 6 présentation et le niveau 7 application.

1.1.1.2 Relations entre couches

On compte deux types de relations intercouches : la relation verticale, pour un même système, et la relation horizontale, entre deux systèmes.

Il y correspond deux types de protocoles : le protocole vertical aussi appelé protocole de service qui permet au niveau (i) de jouir des services du niveau (i-1) et le protocole horizontal qui gère les échanges entre les couches de même niveau.

A chaque couche horizontale correspond un protocole fait de messages paramétrés.

A chaque couche verticale correspond un protocole fait de primitives paramétrées.

On peut définir de manière générale un protocole comme la description des procédures d'échange d'informations entre processus faisant partie d'un réseau d'interconnexion et ayant des fonctions très précises à remplir.

Les fonctions d'un protocole sont assurées par l'échange de messages entre processus. Le format et la signification de ces messages constituent la définition logique et syntaxique du protocole.

Citons à titre d'exemple le protocole LAPB (niveau 2 de la norme X 25).

Une procédure détermine quant à elle les règles d'utilisation séquentielle de communication de messages pour une transmission de données.

Pour remplir ses fonctions, un protocole doit disposer de primitives paramétrées.

Les primitives principales intercouches sont : CON (connection), DISC (disconnection), ACC (acquiescement) et REJ (rejet) pour l'appel et la libération, et, les primitives SEND (envoi) et RECEIVE (réception) pour la transmission de données.

En ce qui concerne les primitives des protocoles horizontaux, on retrouve l'initialisation, la libération, l'envoi de données avec en plus des primitives propres à chaque niveau, afin de remplir leurs fonctions spécifiques.

1.1.2 Fonctionnement général

1.1.2.1 Implémentation du protocole vertical : le moniteur

Une des fonctions fondamentales du moniteur est de gérer la progression intercouche des primitives.

Il existe en fait deux types de progression.

La première est celle de bout en bout, en transit à travers toutes les couches, de deux primitives utilisées pour la connexion et la déconnexion.

La connexion traverse les différentes couches supérieures de l'appelant en testant l'état disponible ou non des couches. Ayant établi une liaison au niveau des couches inférieures, la demande arrive chez l'appelé où le test s'effectuera en montant, jusqu'à l'acceptation de l'utilisateur.

Si tous les tests sont positifs, l'initialisation des différentes couches supérieures suivra, à commencer par la couche numéro 4.

Pour la déconnexion, le mécanisme est similaire, la libération des couches commençant par la couche numéro 6.

Le deuxième type de progression est la progression de primitives à partir de leur niveau (4,5,6,7), traversant les couches qui lui sont inférieures chez l'appelant (l'appelé), puis chez l'appelé (l'appelant), pour aboutir au niveau (4,5,6,7) de son correspondant.

Le moniteur, en décodant les primitives se trouvant dans la zone tampon entre les couches inférieures et les couches supérieures, rend à chaque niveau ce qui lui était destiné. Ce mécanisme concerne l'initialisation, la libération, l'échange des données et des messages (primitives) particuliers.

Il est à remarquer que, pour l'échange de données, le texte transitant dans les couches n'existe qu'en un seul exemplaire.

Des opérations, comme la segmentation, réalisée par les différentes couches, sont effectuées grâce à des points de repère ou pointeurs sur les données.

Parmi les autres fonctions du moniteur, on trouve : la gestion des interruptions et de leurs priorités, la gestion des erreurs de syntaxe et de procédure, ainsi que la gestion des différents temporisateurs (timers) armés par les couches.

1.1.2.2 Gestion du protocole horizontal : l'automate

La gestion interne des couches 4,5,6 est réalisée par des automates d'états.

Ils vérifient le séquençement des actions, assurant ainsi que la procédure se déroule correctement.

1.2 Fonctions des différentes couches

1.2.1 Couche physique

La couche physique fournit les caractéristiques mécaniques, électriques, fonctionnelles et de procédure permettant d'établir, de maintenir et de libérer les connexions entre équipements.

Le niveau physique assure la compatibilité des interfaces physiques afin de permettre la transmission de flots binaires à l'aide d'un câble.

1.2.2 Couche liaison de données

Cette couche a pour but principal l'apport, exempt d'erreurs, des blocs d'informations au niveau supérieur.

Il faut, pour arriver à ce résultat, des services parmi lesquels on trouve :

- détection et contrôle des erreurs de transmission
- contrôle de flux : l'appelant et l'appelé doivent s'arranger pour que l'information n'arrive pas trop vite au récepteur
- séquençement : généralement cette couche doit fournir à la couche supérieure les blocs d'information dans l'ordre où l'autre les a émis.

1.2.3 Couche acheminement (réseau)

La couche réseau a pour rôle de fournir les moyens fonctionnels nécessaires à l'échange des informations fournies par la couche supérieure.

Elle est responsable de l'acheminement des paquets de données.

Pour parvenir à ce résultat, de nombreux services sont réalisés par la couche réseau.

Citons entre autres :

- rendre transparentes les fonctions de commutation et le routage pour le niveau supérieur
- résoudre des problèmes de contrôle d'erreurs et de flux
- l'adressage.

Remarque :

- le service d'établissement de la communication à ce niveau est un service de bout en bout
- les autres services résultent d'un dialogue entre le système et le noeud d'entrée dans le réseau.

1.2.4 Couche transport

La liaison entre les deux correspondants aux niveaux inférieurs étant établie, il reste à assurer l'adressage et l'identification au second degré, celui de l'application dont les programmes d'échange se situeront au niveau immédiatement supérieur, c'est à dire celui de la session.

La fonction première de ce niveau est de contrôler l'établissement et la libération de la connexion, ainsi que les erreurs spécifiques à ce niveau.

C'est un service de bout en bout.

D'autres fonctions sont laissées au choix de l'utilisateur ou de la normalisation.

Parmi celles-ci, les plus importantes sont celles qui doivent rendre l'application de plus en plus indépendante vis-à-vis du réseau, cad : rendre les mêmes services quelque soit la qualité des services offerts par le réseau (faible ou fort).

On pourra donc trouver à ce niveau des fonctions de contrôle que des réseaux dits faibles ne peuvent assurer.

1.2.5 Couche session

La couche session a pour objectif de négocier les interactions entre les utilisateurs pour synchroniser les opérations effectuées sur les données.

Ce niveau doit assurer :

- non seulement la connexion et la libération des entités à ce niveau, mais également le contrôle et les reprises en cas de dérapage au niveau des échanges
- le contrôle de la chronologie et de la qualité des dialogues entre les processus distants de ce niveau. Cela inclut : l'activation, l'initialisation, la synchronisation (et éventuellement la resynchronisation), les échanges, le début et la fin de session.

Ce niveau est composé d'autant de modules d'échanges qu'il y a de types d'applications (transfert de fichiers, interactivité, ...).

1.2.6 Couche présentation

Cette couche a pour but de pouvoir interpréter la signification des données échangées entre les utilisateurs. Elle assure une compréhension syntaxique entre les utilisateurs, en gérant les formats des données à échanger et en effectuant les transformations nécessaires sur les données pour les rendre compréhensibles entre machines hétérogènes.

Les fonctions que doit pouvoir effectuer le niveau sont :

- assurer la connexion et la négociation des paramètres de formatage à ce niveau

- présenter les données échangées entre systèmes dans un format compréhensible par les applications
- rendre les applications indépendantes des modifications d'interface ou d'application d'autres systèmes. Trois grandes catégories de protocoles, correspondant à des applications différentes, sont généralement admises :
 - protocole d'appareil virtuel
 - protocole de fichier virtuel
 - protocole de transfert et de manipulation de travaux.

1.2.7 Couche application

La couche application, qui est la couche la plus externe du modèle de référence de l'ISO, permet la compréhension et l'exécution des commandes liées aux processus d'application.

L'éventail des applications de téléinformatique est très large.

On y trouve des applications dont la liste qui suit n'est pas exhaustive :

- de télétraitement ou encore de ce que l'on appelle le Remote Job Entry
- de type bancaire
- de type conversationnel ou interactif, allant de l'interrogation de bases de données à la réservation de places, de produits, ...
- de type transactionnel
- de transfert de fichiers, de documents
- en temps réel
- du domaine industriel comme la télémessure, la commande de processus industriel.

2 INTERFACES PHYSIQUES

2.1 INTRODUCTION

2.1.1 Définition de l'interface physique

Ce chapitre présente les interfaces physiques généralement utilisées pour la transmission de données par l'intermédiaire du réseau téléphonique, des réseaux spécialisés pour données et par l'intermédiaire des réseaux de télécommunication à intégration de services (RNIS-ISDN) qui outre la parole et les textes supportent la transmission de données.

Une interface peut être définie de manière générale comme point frontière pour lequel sont spécifiées toutes les caractéristiques nécessaires en vue de l'interconnexion et l'interfonctionnement direct de deux équipements de traitement de données ou entre l'équipement terminal de traitement de données et l'équipement de terminaison du circuit de données.

L'interface physique entre équipements est appelé selon la terminologie ISO le niveau physique.

2.1.2 Caractéristiques générales

Les caractéristiques principales sont les suivantes :

- caractéristiques fonctionnelles :
 - nombre de circuits
 - catégorie de circuits
 - fonctions et significations des états logiques
 - définition de l'orientation des circuits
- caractéristiques de procédure :
 - relations entre états logiques
 - définitions des séquences de déroulement
 - définitions des temporisations
- caractéristiques électriques :
 - définition électrique (tensions, type de signalisation, circuits symétriques ou dissymétriques)
 - attribution des valeurs logiques aux valeurs physiques (tension, courants)
 - seuils de fonctionnement
 - forme des signaux
 - schéma de principe
- caractéristiques mécaniques :
 - type de connecteur
 - attribution des fiches (négatives, positives)
 - attribution des broches

2.1.3 Fonctions de l'interface physique

Les fonctions de l'interface physique sont d'offrir au niveau supérieur la possibilité d'établir, de maintenir et de terminer une transmission d'informations binaires sur un circuit.

On peut distinguer quatre phases lors de la transmission : l'établissement de la liaison (nécessaire si le circuit n'est pas relié en permanence), l'initialisation (permet à l'ETCD de s'adapter aux circuits et la synchronisation des bases de temps en mode synchrone), la transmission et la libération.

Remarque : le mode de transmission est dit synchrone lorsque, pour la suite de données à envoyer, l'espace entre deux caractères est connu, dépendant du rythme de l'horloge; une transmission est qualifiée d'asynchrone, d'arythmique ou de start-stop, si, pour la suite de données à envoyer, l'espace entre deux caractères est quelconque; les bits start et stop servent à la reconnaissance du caractère.

2.1.4 Organismes intéressés

Les deux principaux organismes intéressés sont le CCITT pour les caractéristiques fonctionnelles, électriques et de procédure (avec l'ISO), et l'ISO pour les caractéristiques mécaniques.

L'EIA et ANSI ont aussi contribué à l'élaboration de ces avis.

2.2 Interfaces traditionnelles et pour réseau téléphonique

Les réseaux téléphoniques abordés ici sont les réseaux téléphoniques traditionnels.

Les interfaces pour la téléphonie sont celles de la norme V des CCITT.

Nous appelons interfaces traditionnelles celles de type RS de l'EIA.

Un rapprochement entre les deux sera fait lors de ce chapitre.

2.2.1 Interfaces de la série V, introduction

L'interface la plus fréquemment citée est l'interface définie dans la recommandation V 25 associée aux recommandations V 28 ou V 10 - V 11. Elle est généralement désignée simplement par interface V 24.

Ses caractéristiques fonctionnelles sont définies dans la recommandation V 24 elle-même.

Les procédures et modes opératoires relatifs sont pour l'essentiel également contenus dans la même recommandation. Des spécifications de détail (circuits effectivement utilisés, valeurs de temporisation) sont toutefois intégrées directement dans les recommandations qui décrivent les modems.

De plus, le cas particulier des procédures pour l'appel et la réponse automatique sur le réseau téléphonique commuté est traité dans la recommandation V 25.

Les caractéristiques électriques sont décrites dans la recommandation V 28, V 10 - V 11, ainsi que V 31 et V 35 pour les cas particuliers.

Les caractéristiques mécaniques sont définies dans la norme ISO 2110 (description du connecteur et affectation des broches) lors de l'utilisation des caractéristiques électriques selon V 28, et, dans la norme ISO 4902 pour l'application de V 10 - V 11.

2.2.2 Recommandation V 24

2.2.2.1 Vue générale des circuits et comparaison RS

La recommandation V 24 prévoit au total 55 circuits de jonction possibles entre un ETTD et un ETCD (modem). Seul un sous-ensemble est généralement utilisé en fonction du type de modem et du mode d'exploitation envisagé.

Les circuits de jonction se répartissent en deux groupes :

- les circuits de la série 100 qui relient le modem à l'ETTD
- les circuits de la série 200 qui permettent de connecter un ETTD à un dispositif d'appel automatique; dans ce dernier cas, l'ETCD se compose du modem et du dispositif d'appel.

En outre, on fait une distinction entre cinq catégories fonctionnelles de circuits :

- circuits de terre
- circuits de données
- circuits de base de temps
- circuits de commandes
- circuits de confirmation de commande et d'annonce d'état.

Une seule fonction est attribuée à chaque circuit.

Parmi les 55 circuits disponibles, 10 à 20 d'entre eux environ sont employés pour un modem particulier.

Les circuits les plus importants entre un terminal (T) et un modem (M) sont :

- circuits de données :
 - 103 : émission des données (T→M).
Les signaux data sont envoyés par le terminal vers le modem et sont destinés à la station éloignée.
 - 104 : réception des données (M→T).
Les signaux data du modem vers le terminal arrivent de la station éloignée.
- circuits de base de temps : (synchrone uniquement)
 - 114 : base de temps à l'émission (M→T).
Horloge d'émission du modem, envoyée vers le terminal pour l'émission de ses données. Le terminal doit fournir un signal via le 103 dans lequel les transitions entre éléments (bits) se produisent en même temps que les transitions de l'état ouvert à l'état fermé du 114.
 - 115 : base de temps à la réception (M→T).
Le modem transmet l'horloge de réception des data de réception au terminal. La transition de l'état fermé à l'état ouvert indique le milieu d'un bit data sur la liaison 104.

- circuits de commande :
 - 108/1 : connectez le poste de données sur la ligne (T-->M).
Le signal de contrôle commande le raccordement du modem à la ligne.
Etat fermé = raccordez le modem à la ligne.
 - 105 : demande pour émettre (T-->M).
Le signal de contrôle arrivant du terminal demande d'envoyer un signal de ligne au modem afin de réaliser le circuit data.
L'état fermé raccorde l'émetteur.
- circuits de confirmation de commande ou d'annonce d'état :
 - 107 : poste de données prêt (M-->T).
Signal de contrôle qui annonce que le modem est raccordé à la ligne de transmission.
Etat fermé = modem prêt.
 - 106 : prêt à émettre (M-->T).
Signal de contrôle qui annonce au terminal que la liaison de données est construite et que les data peuvent être envoyées.
Etat fermé = prêt à émettre.
 - 109 : détecteur du signal reçu sur la voie de données (M-->T).
Le modem annonce au terminal qu'un signal de ligne est reçu à un niveau au dessus d'un seuil préétabli.
Etat fermé : le signal est reçu à un niveau supérieur au seuil.
- circuits de terre :
 - 102 : terre de signalisation ou retour commun (masse).
Ce conducteur établit le retour commun pour les circuits de jonction dissymétriques avec des caractéristiques électriques conformes à l'avis V 28 et le potentiel continu de référence pour les circuits symétriques conformes à l'avis V 35.

| Circuit de jonction N° | Désignation du circuit de jonction | Sens |
|------------------------|---|-------|
| 102 | Terre de signalisation ou retour commun | |
| 102A | Retour commun ETTD | |
| 102B | Retour commun ETCD | |
| 102C | Retour commun | |
| 103 | Emission des données | T → M |
| 104 | Réception des données | T ← M |
| 105 | Demande pour émettre | T → M |
| 106 | Prêt à émettre | T ← M |
| 107 | Poste de données prêt | T ← M |
| 108/1 | Connectez le poste de données sur la ligne | T → M |
| 108/2 | Equipement terminal de données prêt | T → M |
| 109 | Détecteur du signal de ligne reçu sur la voie de données | T → M |
| 110 | Détecteur de la qualité du signal de données | T ← M |
| 111 | Sélecteur du débit binaire (ETTD) | T → M |
| 112 | Sélecteur du débit binaire (ETTD) | T ← M |
| 113 | Base de temps pour les éléments de signal à l'émission (ETTD) | T → M |
| 114 | Base de temps pour les éléments de signal à l'émission (ETCD) | T ← M |
| 115 | Base de temps pour les éléments de signal à la réception (ETCD) | T ← M |
| 116 | Choix d'organes en réserve | T → M |
| 117 | Indicateur mode en réserve prêt | T ← M |
| 118 | Emission de données sur la voie de retour | T → M |
| 119 | Réception de données sur la voie de retour | T ← M |

Figure 2.1: Liste des circuits.

| Circuit de jonction N° | Désignation du circuit de jonction | Sens |
|------------------------|---|-------|
| 120 | Transmettre les signaux de ligne sur la voie de retour | T → M |
| 121 | Voie de retour prête | T ← M |
| 122 | Détecteur de signal reçu en ligne sur la voie de retour | T ← M |
| 123 | Détecteur de la qualité du signal sur la voie de retour | T ← M |
| 124 | Choix des groupes de fréquences | T → M |
| 125 | Indicateur d'appel | T ← M |
| 126 | Choix de la fréquence d'émission | T → M |
| 127 | Choix de la fréquence de réception | T → M |
| 128 | Base de temps pour les éléments de signal à la réception (ETTD) | T → M |
| 129 | Demande pour recevoir | T → M |
| 130 | Transmettre la tonalité sur la voie de retour | T → M |
| 131 | Base de temps pour les caractères reçus | T ← M |
| 132 | Retour au mode (pas pour données) | T → M |
| 133 | Prêt à recevoir | T → M |
| 134 | Données reçues présentes | T ← M |
| 136 | Nouveau signal | T → M |
| 140 | Bouclage/essai de maintenance | T → M |
| 141 | Bouclage local | T → M |
| 142 | Indicateur d'essai | T ← M |

Figure 2.2: Liste des circuits (suite)

| Circuit de jonction N° | Désignation du circuit de jonction | Sens |
|------------------------|---|-------|
| 191 | Emission de la réponse à la voix | T → M |
| 192 | Réception de la réponse à la voix | T ← M |
| 201 | Terre de signalisation ou retour commun | |
| 202 | Demande d'appel | T → M |
| 203 | Ligne pour données occupée | T ← M |
| 204 | Poste éloigné connecté | T ← M |
| 205 | Abandon de l'appel | T ← M |
| 206 | Signal numérique | T → M |
| 207 | Signal numérique | T → M |
| 208 | Signal numérique | T → M |
| 209 | Signal numérique | T → M |
| 210 | Présentez le chiffre suivant | T ← M |
| 211 | Chiffre présent | T → M |
| 213 | Indication d'alimentation | T ← M |

Figure 2.3: Liste des circuits (suite et fin)

| CCITT Recommendation V.24 | | EIA RS.449 | | EIA RS.232.C | |
|---------------------------|--|------------|--------------------------|--------------|--|
| 102 | Signal ground | SG | Signal ground | AB | Signal ground |
| 102a | DTE common | SC | Send common | | |
| 102b | DCE common | RC | Receive common | | |
| 125 | Calling indicator | IS | Terminal in service | | |
| 108/2 | Data terminal ready | IC | Incomming call | CE | Ring indicator |
| 107 | Data set ready | TR | Terminal ready | CD | Data terminal ready |
| | | DM | Data mode | CC | Data set ready |
| 103 | Transmitted data | SD | Send data | BA | Transmitted data |
| 104 | Received data | RD | Receive data | BB | Received data |
| 113 | Transmitter signal element timing (DTE source) | TT | Terminal timing | DA | Transmitter signal elem. timing (DTE source) |
| 114 | Transmitter signal element timing (DCE source) | ST | Send timing | DB | Transmitter signal elem. timing (DCE source) |
| 115 | Receiver signal element timing (DCE source) | RT | Receive timing | DD | Receiver signal element timing |
| 105 | Request to send | RS | Request to send | CA | Request to send |
| 106 | Ready for sending | CS | Clear to send | CB | Clear to send |
| 109 | Data channel received line signal detector | RR | Receiver ready | CF | Received line signal detector |
| 110 | Data signal quality detector | SQ | Signal quality | CG | Signal quality detector |
| | | NS | New signal | | |
| 126 | Select transmit frequency | SF | Select frequency | | |
| 111 | Data signaling rate selector (DTE source) | SR | Signaling rate selector | CH | Data signal rate selector (DTE source) |
| 112 | Data signaling rate selector (DCE source) | SI | Signaling rate indicator | CI | Data signal rate selector (DCE source) |

Figure 2.4: Tableau de comparaison.

| CCITT Recommendation V.24 | | EIA RS.449 | | EIA RS.232.C | |
|---------------------------|--|------------|---------------------------|--------------|---|
| 118 | Transmitted backward channel data | SSD | Secondary send data | SBA | Secondary transmitted data |
| 119 | Received backward channel data | SRD | Secondary receive data | SBB | Secondary received data |
| 120 | Transmit backward channel line signal | SRS | Secondary request to send | SCA | Secondary request to send |
| 121 | Backward channel ready | SCS | Secondary clear to send | SCB | Secondary clear to send |
| 122 | Backward channel received line signal detector | SRR | Secondary receiver ready | SCF | secondary received line signal detector |
| 141 | Local loopback | LL | Local loopback | | |
| 140 | Remote loopback | RL | Remote loopback | | |
| 142 | Test indicator | TM | Test mode | | |
| 116 | Select standby | SS | Select standby | | |
| 117 | Standby indicator | SB | Standby indicator | | |

Figure 2.5: Tableau de comparaison (suite)

2.2.2.2 Caractéristiques électriques de l'interface V24.

Il existe cinq normes différentes pour fixer les caractéristiques électriques d'une interface de la série V. Elles sont contenues dans les recommandations V 28, V 10 et V 11, ainsi que dans V 31 et V 35.

2.2.2.2.1 Norme V 28

Au début des années 1980, la norme V 28 était la plus utilisée.

La combinaison V 24 - V 28 correspond pour l'essentiel à la norme RS-232-C établie par l'EIA.

Il existe cependant certaines différences touchant les types de circuits de jonction disponibles et leur terminologie.

La norme V 28 impose des restrictions concernant le débit binaire entre ETTD et ETCD (20 k b/s maximum) et la longueur du câble de connexion (15 m selon l'EIA).

2.2.2.2.2 Normes V 10 / V 11

Ces recommandations ont été établies pour que la réalisation des interfaces, à l'aide de composants intégrés, soit simplifiée.

De plus, l'objectif fixé visait à éliminer les restrictions de débit binaire et de distance de câble imposées par V 28.

L'interface V 10, à utiliser pour des débits binaires allant jusqu'à 100 k b/s, est comparable à l'interface V 28. L'interface V 11, à utiliser pour des débits binaires allant jusqu'à 10 M b/s, est comparable à l'interface V 35.

La longueur des câbles maximale varie avec le débit utilisé, à titre d'exemple : avec 100 mètres de câble, on va jusqu'à 9600 b/s avec V 10 et 1 M b/s avec V 11.

2.2.2.2.3 Normes V 31 et V 35

Les jonctions définies dans les recommandations V 31 et V 35 sont citées à titre indicatif.

Elle n'ont pas trouvé une large diffusion et doivent être considérées comme des solutions transitoires.

| | Débit binaire maximum | Type des circuits | Attribution logique des signaux | Niveaux significatifs |
|------|-----------------------------|----------------------|--|------------------------------|
| V.28 | 20 K b/s | Dissymétrique | Tension Négative=1 ou ouvert Positive=0 ou fermé | $\leq -3V$ $\geq +3V$ |
| V.10 | 100 K b/s | Dissymétrique | Tension Négative=1 ou ouvert Positive=0 ou fermé | $\leq -0,3V$ $\geq +0,3V$ |
| V.11 | 10 M b/s | Symétrique | Tension Négative=1 ou ouvert Positive=0 ou fermé | $\leq -0,3V$ $\geq +0,3V$ |
| V.35 | 48 K b/s | Symétrique | Tension Négative=1 ou ouvert Positive=0 ou fermé | pas spécifié |
| V.31 | 75 b/s | Symétrique | Contact fermé=1 Contact ouvert=0 | 0,1...15 mA |

Figure 2.6: Tableau des caractéristiques électriques.

2.2.2.3 Caractéristiques mécaniques

Lorsque la combinaison V 24 / V 28 est appliquée, le connecteur utilisé comprend 25 broches. Il est spécifié dans la norme ISO 2110.

Pour l'application de V 10 / V 11, il est recommandé en revanche d'utiliser un connecteur de 37 broches, défini dans la norme ISO 4902.

Le fait de choisir 37 broches doit permettre de supprimer les limitations imposées par les 25 broches du premier connecteur, qui empêchent une affectation univoque des circuits de fonction V 24.

Lorsque le connecteur à 37 broches est insuffisant (en particulier lorsque les numéros 118 à 123 prévus pour la voie de retour sont utilisés), un connecteur à 9 broches peut compléter celui qui en comprend 37.

La combinaison V10/V11/V24/ISO 4902 correspond à la norme américaine RS-449 de l'EIA.

Pour la combinaison V 35/ V 24, on utilise un connecteur à 34 broches.

2.2.3 Les modems

2.2.3.1 Introduction

Les signaux provenant de l'ETTD peuvent uniquement accepter deux états donnés par 0 et 1 auxquels corresponde un niveau électrique particulier.

Le train d'impulsions qui existe à l'interface par l'envoi des bits, est appelé signal bande de base ou signal NRZ (non return to zero).

Vu les restrictions imposées par la technique de transmission (distorsions, bruit, atténuation, ...), le signal est transformé au moyen d'un modem afin d'être émis sous une forme adaptée.

Selon le type de transmission, on peut répartir les modems comme suit :

- modems pour transmission dans la bande téléphonique 300 - 3400 hz.
Ceux-ci modulent le signal bande de base en un signal analogique.
Ils sont appelés modems bande téléphonique, modems VF ou modems classiques.
Ils peuvent utiliser deux types de support : le réseau téléphonique avec un débit maximum de 4800 b/s ou alors des liaisons fixes, appelées aussi lignes téléphoniques spécialisées (avis M 1020), liaisons louées ou circuits d'abonnement avec un débit binaire maximum de 14400 b/s.
- les modems pour la transmission dans la bande groupe primaire 60 - 108 k hz, définie par l'avis H 14.
Le signal bande de base est codé puis modulé.
Ils sont appelés modems de bande de groupe ou modems à larges bandes.
Les débits binaires sont de 48, 56, 64 ou 72 k b/s.
- les modems pour transmission sur distances limitées via des liaisons physiques purement zonales.
Ils codifient le signal bande de base vers une forme plus adaptée à la transmission. La transmission reste digitale.
Ils ne font pas encore l'objet d'une normalisation des CCITT.
- les coupleurs acoustiques sont des modems utilisant le poste téléphonique habituel.
L'avis V 15 des CCITT traite de l'utilisation de ces appareils pour la transmission de données.
- les modems pour applications spéciales :
 - les modems V 16 pour la transmission de données analogiques médicales (par exemple : électrocardiogramme).

- les modems V 19 pour transmission parallèle de données utilisant les fréquences de signalisation des postes téléphoniques à clavier.
- les modems V 20 pour la transmission de données parallèle via le réseau public commuté.

Les différents débits cités ont aussi fait l'objet d'une normalisation : l'avis V 5 pour le réseau téléphonique commuté et l'avis V 6 pour les lignes louées de type téléphonique.

2.2.3.2 Liste des modems et de leurs caractéristiques

La liste des modems repris dans ce chapitre n'est pas exhaustive.

2.2.3.2.1 Quelques considérations plus techniques

Le débit binaire (b/s) est l'unité de la transmission synchrone.

Pour la transmission asynchrone, vu entre autre l'incertitude qui existe sur la durée entre caractères STOP, on préfère la notion de rapidité de modulation (bauds).

La relation entre le débit binaire (db) et la vitesse de modulation (vm) dépendra du nombre d'états significatifs (q) que pourra prendre le signal :

$$db = vm \log_2 q$$

Le groupage de bits n'est possible que si le signal émis peut prendre plusieurs états significatifs.

Il est la plupart du temps utilisé pour les modems synchrones.

Il est à remarquer que, malgré tout, certains parlent de débit binaire pour les transmissions asynchrones : ils devraient préciser ce qu'ils entendent par là.

Les modulations généralement utilisées sont les modulations de fréquences pour les basses vitesses et celles de phases, de phases et d'amplitudes pour les grandes vitesses.

2.2.3.2.2 Liste des modems

La liste des modems est reprise dans les tableaux des figures 2.7 et 2.8.

2.3 Interfaces pour réseaux publics pour données

2.3.1 Introduction

Les interfaces de la série X ont été normalisés par le CCITT avec la spécification des nouveaux réseaux publics spécialisés pour données, à commutation de circuits ou par paquets.

Un des objectifs visés consistait dans l'obtention d'une interface physique moins complexe que V 24 et comportant, en particulier, un nombre plus réduit de circuits, tout en exploitant les avantages des caractéristiques électriques déjà définies pour l'interface V 24.

Les caractéristiques fonctionnelles des interfaces de la série X sont spécifiées dans la recommandation X 24.

Les procédures des interfaces de la série X ne sont esquissées que très partiellement dans X 24.

Les procédures sont spécifiées dans les recommandations dédiées aux applications spécifiques de l'interface. Il s'agit en particulier des applications suivantes :

- les réseaux publics de données à commutation de circuits avec transmission arithmique, selon la recommandation X 20;
- les réseaux publics de données à commutation de circuits pour fonctionnement synchrone, selon la recommandation X 21;
- les réseaux publics de données à commutation par paquets, selon la recommandation X 25;
- la recommandation X 21 prévoit également, comme cas particulier, que l'interface X 24 peut être appliquée à l'exploitation de circuits loués.

Chacune des recommandations citées ci-dessus contient les spécifications de détail des interfaces physiques entre ETTD et ETCD.

X 20 et X 21 parlent d'interface physique et X 25 de niveau physique.

Les caractéristiques électriques des interfaces de la série X sont contenues dans les recommandations X 26 et X 27.

En fait, ces recommandations renvoient au texte des recommandations V 10 et V 11 (X 26 = V 10 et X 27 = V 11).

Les caractéristiques mécaniques sont définies dans la norme ISO 4903, tant pour les services X 20 et X 21 que pour X 25.

Le connecteur possède 15 broches.

Remarque : il est à noter que, lorsqu'on parle d'ETCD dans

| Adresse CCITT | Débit binaire b/s | Vitesse de modulation baud | Mode | Modulation | Canal auxi- liaire | Avis corres- pondant | Sup- port |
|------------------|-------------------------|----------------------------------|-------------------|----------------------------------|--------------------------|----------------------------|--------------|
| V21 | 200 (300) | 200 (300) | ASYN | FM, 2 voies | | V24 et V28 | LS |
| V23 | 600 1200 | 600 1200 | SYN et ASYN | FM FM | X X | V24 et V28 | RC LS |
| V26 | 2400 1200 | 1200 1200 | SYN | PM, 4 phases PM, 2 phases | X X | V24 et V28 | LS |
| V26 bis | 2400 1200 | 1200 1200 | SYN | PM, 4 phases PM, 2 phases | X X | V24 et V28 | RC |
| V27 | 4800 2400 | 1600 1200 | SYN | PM, 8 phases PM, 4 phases | X X | V24 et V28 | LS |
| V27 bis | 4800 2400 | 1600 1200 | SYN | PM, 8 phases PM, 4 phases | X X | V24 et V28 | LS |
| V27 ter | 4800 2400 | 1600 1200 | SYN | PM, 8 phases PM, 4 phases | X X | V24 et V28 | RC |

Figure 2.7: Tableau des modems.

| Adresse CCITT | Débit binaire b/s | Vitesse de modulation baud | Mode | Modulation | Canal auxi- liaire | Avis corres- pondant | Sup- port |
|------------------|-------------------------|----------------------------------|------|---------------------------|--------------------------|----------------------------|--------------|
| V29 | 9600 | 2400 | SYN | AM, 8 phases 2 niveaux | | V24 | LS |
| | 7200 | 2400 | | PM, 8 phases | | et | |
| | 4800 | 2400 | | PM, 4 phases | | V28 | |
| V35 | 48 K | | SYN | AM | | V24 V10 V11 | GP |
| V36 | 48 K | | SYN | AM | | V24 V10 V11 | GP |
| | 56 K | | | | | | |
| | 64 K | | | | | | |
| | 72 K | | | | | | |

FM modulation de fréquence

RC réseau commuté

PM modulation de phase

LS lignes spécialisées

AM modulation d'amplitude

GP groupe primaire

Figure 2.8: Tableau des modems (suite)

la norme X, il s'agit du noeud d'entrée dans le réseau.

2.3.2 La recommandation X 24

La recommandation X 24 prévoit au total 10 circuits de jonction entre l'ETTD et l'ETCD, à savoir :

- 3 circuits de terre (G, Ga, Gb)
- 2 circuits de données (T, R)
- 2 circuits de commande (C, I)
- 2 circuit de base de temps (S, B)
- 1 circuit de reconnaissance de début de trame (F).

Les circuits de données peuvent aussi bien être utilisés pour la transmission de commandes, par exemple lors de l'établissement ou de la libération d'une liaison commutée.

Le nombre et le type de circuit utilisés dépendent de l'application (X20, X21, X25).

2.3.3 Interface physique selon X 20

L'interface X 20 ne comprend que les circuits de données à l'émission et à la réception (T et R), ainsi que les trois circuits de terre (G, Ga, Gb).

Les caractéristiques électriques correspondent à la recommandation X 26 du côté ETCD. Côté ETTD, X 26, X 27 ou V 28 peuvent être appliquées.

Le connecteur est celui qui comprend 15 broches selon la norme ISO 4903.

2.3.4 Interface physique selon X 21

L'interface X 21 comprend les circuits de données (T et R), deux circuits de terre (G, Ga), les circuits de commande et d'indication (C et I) et un circuit de base de temps (S). Une base de temps pour les multiplets (octets) (B) peut être fournie à titre additionnel.

Les caractéristiques électriques du côté ETCD correspondent à la recommandation X 27.

Côté ETTD, X 26 ou X 27 peuvent être appliquées pour un débit inférieur ou égal à 9600 b/s. Au delà, seul X 27 est autorisée.

Les caractéristiques mécaniques sont conformes à la norme ISO 4903.

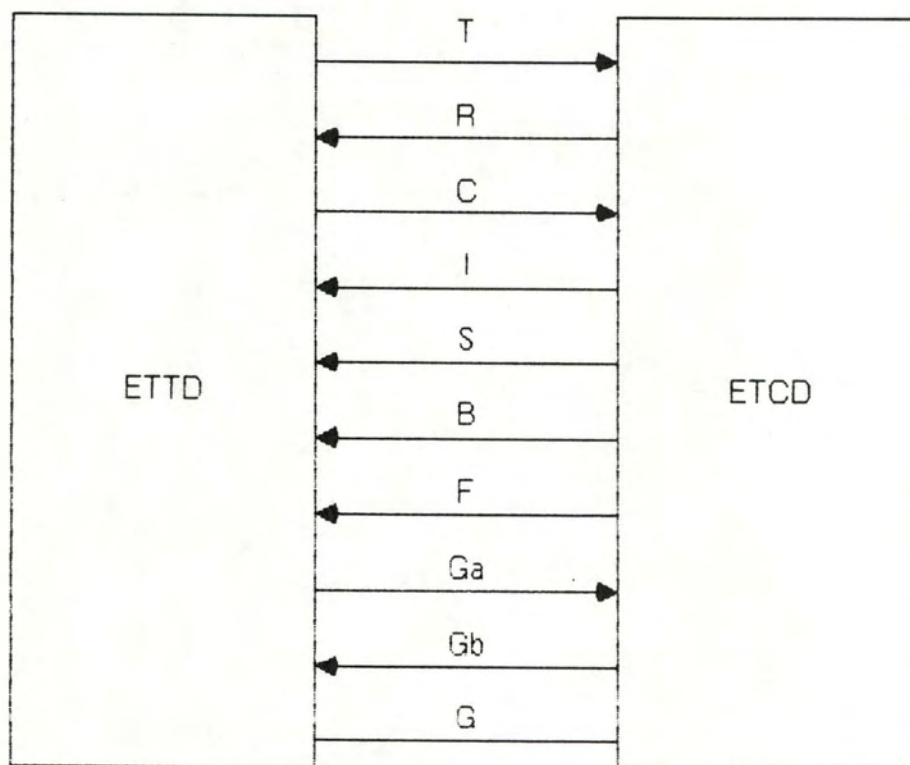


Figure 2.9: Tableaux des circuits X24

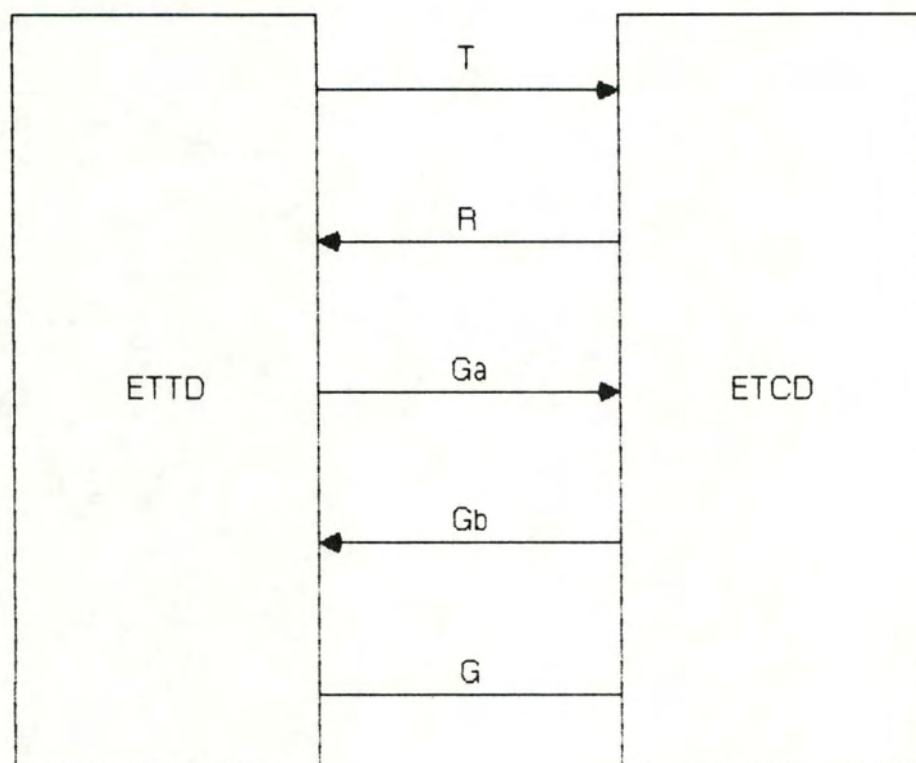


Figure 2.10: Tableau des circuits X20

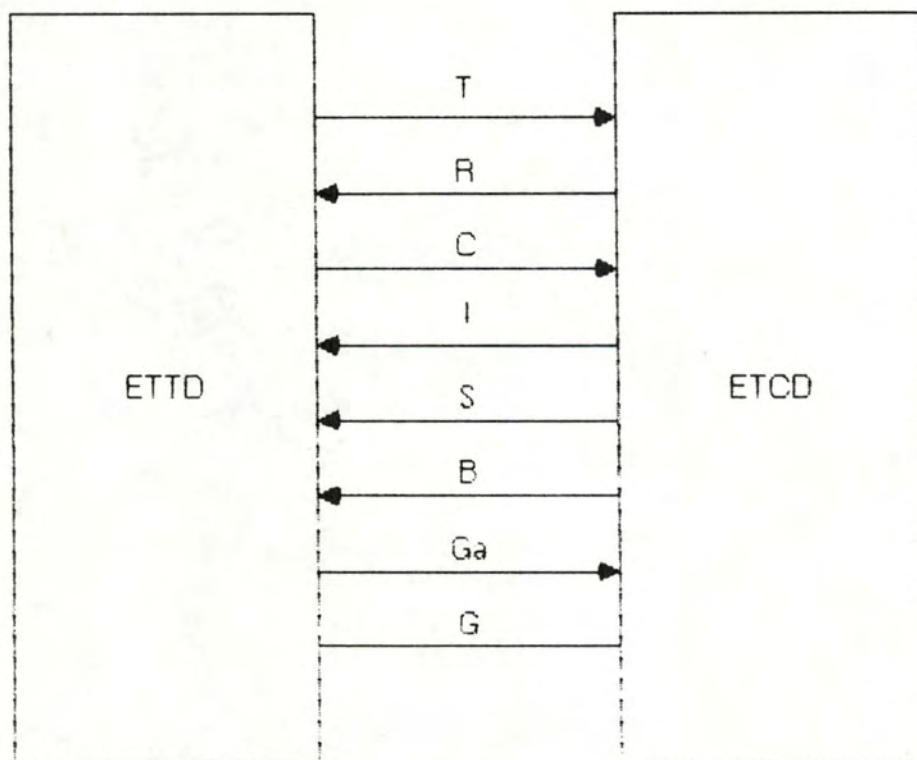


Figure 2.11: Tableau des circuits X21

2.3.5 Interfaces physiques pour X 25

2.3.5.1 Mode synchrone

L'interface physique décrit dans la recommandation X 25 est identique à celle qui est contenue dans la recommandation X 21.

2.3.5.2 Mode asynchrone

La recommandation X 24 décrit l'interface ETTD/ETCD pour l'accès d'un ETTD arythmique au service complémentaire d'assemblage et de désassemblage de paquets (ADP) dans un réseau public pour données situé dans le même pays.

La recommandation X 3 décrit le service complémentaire d'assemblage et de désassemblage de paquets (ADP).

La recommandation X 29 décrit les procédures d'échanges de l'information de commande et des données de l'utilisateur entre un ETTD fonctionnant en mode paquet (synchrone) et un ADP.

Les fonctions et procédures sont spécifiées dans ces trois recommandations.

L'interface comprend 20 paramètres modifiables par l'utilisateur.

Ces paramètres définissent les caractéristiques du terminal et le format des données à afficher.

L'interface entre le modem et l'ETTD est l'interface V 24.

Les circuits utilisés sont les circuits 103, 104, 105, 106, 107, 109, 108.2, 125 et 142.

2.3.6 Interfaces X 20 bis et X 21 bis

Lors de l'étude et de la mise en oeuvre des réseaux commutés de données spécialisés, il s'est révélé que la non disponibilité immédiate d'ETTD équipés des interfaces selon les recommandations X 24/ X 26/ X 27 constituait un handicap au développement rapide des nouveaux réseaux publics. En outre, les utilisateurs désiraient conserver le parc de terminaux et d'ordinateurs existants pour lequel ils avaient consenti des investissements.

Cette situation a amené le CCITT à définir deux recommandations à objectif intérimaire, X 20 bis et X 21 bis, permettant d'utiliser des ETCD et ETTD équipés d'une interface V 24 dans le cadre des procédures physiques spécifiées pour les services X 20, X 21 et X 25.

Cette solution est à double tranchant.

D'une part, elle favorise la phase initiale de développement des réseaux publics.

D'autre part, elle peut freiner les initiatives de l'industrie souhaitant introduire les nouvelles interfaces dans les équipements, du fait que la pression des utilisateurs sur les constructeurs est moins forte.

Le tableau qui suit précise quels sont les circuits de l'interface V 24 qui sont appliqués pour le service X 21 dans la version X 21 bis.

Dans le cas particulier de l'utilisation de l'interface physique X 21 bis pour l'interface physique X 25, les circuits V 24 de commande numéro 105, 106, 107, 108 et 109 sont utilisés et doivent être à l'état fermé, pour que les procédures d'ordre supérieur (niveaux 2 et 3) puissent se dérouler correctement à travers les circuits 103 et 104.

Circuits V.24 utilisés : N° 102, 103, 104, 105, 106, 107, 108/1

ou 108/2, 109, 114, 115, 125, 141, 142

| Etat de la communication | Etat de la jonction | | | | | | |
|--------------------------|---------------------------|-----|-----|-------|-------|-----|-----|
| | O = ouvert F = fermé | | | | | | |
| | Circuit N° | | | | | | |
| | 105 | 106 | 107 | 108/1 | 108/2 | 109 | 125 |
| Prêt | F | F | F | F | F | F | O |
| Demande d'appel | F | F | F | F | — | F | O |
| Appel entrant | F | F | O | O | O | F | F |
| Acceptation de l'appel | F | F | O | F | F | F | O |
| Prêt pour données | F | F | F | F | F | F | O |
| Demande libération | F | F | F | O | O | F | O |
| Confirmation libération | | | | | | | |
| ETCD | F | F | O | O | O | F | O |
| Indication libération | F | F | O | F | F | F | O |
| Confirmation libération | | | | | | | |
| ETTD | F | F | O | O | O | F | O |
| Transfert des données | F | F | F | F | F | F | O |

Figure 2.12: Tableaux des circuits V 24 utilisés.

2.4 Interfaces pour le RNIS

Pour les niveaux 2 et 3 (ISO), on trouve les interfaces R, S et T.

Elles sont semblables, au point de vue fonctionnel, à leur correspondantes dans X 25.

L'interface R se situe entre le terminal et l'adaptateur au réseau RNIS.

L'interface S se situe entre l'adaptateur et la régie d'abonnés.

La trame est constituée des parties B et D, B étant destiné à l'information et D à la signalisation.

L'interface T se trouve dans la régie d'abonnés. Elle réalise le multiplexage sur la voie de sortie vers la ligne RNIS.

Au niveau 1, interface physique, on trouve l'interface U qui se situe entre la régie d'abonnés et le noeud d'entrée dans le réseau.

Ce niveau est principalement constitué d'une puce HDLC, générant les fanions, gérant le contrôle à l'arrivée et le contrôle du CRC.

Voici les principales fonctions de l'interface U :

- génération et contrôle du HDB 3
- multiplexage et démultiplexage
- contrôle de la synchronisation et du verrouillage
- supprimeur d'écho.

Il est à noter que seules les interfaces R, S et T sont normalisées.

3 PROTOCOLES DE LIAISON DE DONNEES

3.1 Introduction

Le niveau physique permet la connexion, le transfert et la libération sur un circuit de données.

Lors du transfert, de nombreuses erreurs de communication se produisent sur la ligne.

Le but principal du niveau liaison de données est de récupérer ces erreurs, à savoir : les détecter et les contrôler.

Pour cela, il existe des protocoles qui assurent le contrôle d'erreurs.

De par leurs procédures, ils permettent aussi un contrôle de flux.

On distingue généralement deux types de procédure : la procédure 'stop and wait' et la procédure à fenêtre fixe ou glissante.

On distingue aussi deux types de protocoles : les protocoles orientés caractères et ceux orientés bits.

3.2 Le protocole BSC

3.2.1 Introduction

Le protocole BSC ou BYSINC fut un des premiers protocoles développés par IBM.

C'est un protocole orienté caractères, servant, entre autre, au 'polling remote terminal'.

Il est employé sur une ligne half duplex (le full duplex n'étant pas nécessaire), multipoint ou point à point.

Il supporte trois ensembles de caractères : ASCII, EBCDIC et le 6-bit transcode.

La procédure employée est le 'stop and wait'.

3.2.2 Relation entre partenaires

Les statuts des différents partenaires, lors de l'échange d'informations, sont des statuts temporaires.

Sur une ligne point à point, c'est une relation maître-esclave.

La station qui envoie les données prend le statut de maître, et celle qui reçoit, celui d'esclave.

Sur une ligne multipoint, la station qui gère le dialogue en envoyant des invitations à émettre aux autres stations est la station primaire.

Les autres sont appelées secondaires ou tributaires.

Ce sont des statuts fixes.

Lors de l'échange d'informations, on retrouve la relation temporaire maître-esclave.

Les autres stations, inactives à ce moment, reçoivent le statut de neutre.

3.2.3 Procédure 'stop and wait'

Cette procédure est basée sur le principe suivant : à tout message envoyé par le maître, correspond un message de confirmation de réception ou acquittement de la part de l'esclave que le maître attend avant de passer au message suivant.

Ceci empêche bien entendu le transfert continu d'informations entre les deux correspondants.

De plus, puisqu'ils ne parlent jamais en même temps, une ligne half duplex est suffisante.

Il est à noter que la procédure employée par BSC utilise des acquittements numérotés modulo deux.

Ceci permet la détection de la perte d'un acquittement.

3.2.4 Réalisation de la transparence

La transparence, c'est le fait de pouvoir transférer n'importe quel type d'information.

Il s'agit en fait de supprimer toute interférence possible avec le protocole utilisé.

La transparence est utilisée, par exemple, pour la transmission de fichiers binaires.

La technique utilisée pour ce genre de protocole s'appelle le 'character stuffing'.

Le principe est le suivant :

quand une séquence de contrôle apparaît dans la séquence de données à l'émission, le niveau 1 insère un caractère supplémentaire de contrôle que le correspondant supprimera avant de restituer les données à son niveau 2.

Remarque : ce genre de technique de réalisation de la transparence n'est pas idéale car elle est dépendante du code-caractère employé.

3.2.5 Messages utilisés

3.2.5.1 Format des données

L'élément d'information est la trame de caractères, séparée par des caractères de synchronisation.

Le format d'une trame de données est le suivant:

| | | | | | | | |
|-----|-----|-----|--------|-----|------|------------------|--------------|
| SYN | SYN | SOH | Header | STX | Data | ETB or ETX | Check sum |
|-----|-----|-----|--------|-----|------|------------------|--------------|

Figure 3.1: Format d'une trame de données.

- SYN : caractère de synchronisation.
- SOH : caractère signifiant le début de l'entête (header).
- La contenance du header n'est pas définie par le protocole (pour le niveau réseau).
- STX : caractère signifiant le début des données.
- Data : zone contenant les données.
L'unité de données est le bloc, de longueur à définir.
Les records seront répartis dans ces blocs.
La transmission peut être monobloc ou multibloc.
- ETB : termine le bloc (quand plusieurs blocs).
- ETX : termine le dernier bloc.

Remarques :

- Quand on est en multipoint, l'adresse ne se trouve pas dans le header mais dans un message de contrôle.
- En ASCII, le 'Check sum' est un 'vertical parity check'.
Pour les autres codes, les caractères individuels ne subissent pas de vérification de parité. On utilise plutôt le CRC (Cyclic Redundancy Checksums).

3.2.5.2 Séquence de supervision

La supervision est basée sur la notion d'invitation ou polling.

3.2.5.2.1 Définition des méthodes de polling

- Polling : invitation à émettre.
- Selecting : invitation à recevoir.
- Roll call polling : le maître envoie une invitation à émettre à chaque terminal, un par un.
- Hub polling : le maître donne une invitation à émettre au terminal le plus éloigné, qui, s'il n'a rien à émettre, la passe à son voisin; ceci évite de renverser la ligne pour rien.

3.2.5.2.2 Éléments de séquence du BSC

EOT : End Of Transmission (remise à zéro).

ENQ : Enquiry (invitation).

ACK : Acknowledgement (acquiescement).

NACK : Negative acknowledgement (refus).

3.2.6 Remarques générales

Parmi les problèmes que l'on rencontre lors de l'utilisation de ce genre de protocole, apparaissent la longueur du caractère et la performance.

La longueur du caractère est fixe, soit 6, soit 8 pour BYSINC.

Tout code utilisant 9 comme longueur de caractère ne peut être employé.

Les performances de ce genre de protocole sont relativement faibles : en moyenne, dans des conditions normales d'utilisation, on obtient un débit utile égal à la moitié du débit binaire.

Il importe donc de trouver des moyens permettant d'augmenter le débit utile.

Parmi ces moyens, on trouve :

- si la ligne est de meilleure qualité que prévu :
 - l'augmentation de la longueur des blocs
 - la suppression de l'`intermediate block checking`, pour éviter les retournements fréquents du sens de transmission dus aux acquittements des blocs d'informations.
- en conditions normales, la suppression des blancs :
 - compresser toute suite de deux blancs ou plus
 - tronquer la dernière suite de blancs
- pour une ligne multipoint :
 - utilisation de la technique du `hub polling`.

3.3 Les protocoles évolués

3.3.1 Introduction

Les protocoles orientés caractères sont des protocoles de l'ancienne génération, dont le principal défaut est la lenteur.

La nouvelle génération de protocole commença dans les années septante, avec le développement par IBM du protocole SDLC, Synchronous Data Link Control, pour son réseau SNA.

En reprenant les idées principales du SDLC, ANSI développait le ADCCP, Advanced Data Communication Control Procedure, tandis que l'ISO produisait le HDLC, High level Data Link Control.

Le CCITT se basa sur le HDLC pour établir le niveau 2 de X 25, à savoir : le LAP, Link Access Procedure. La nouvelle version en date, le LAPB, est utilisée entre autre par les PTT suisses et belges.

Ces nouveaux protocoles sont orientés bits. Le nombre de bits dans les trames de données est arbitraire et non plus un multiple d'une longueur de caractère spécifique.

De plus, avec leur procédure à fenêtre glissante orientée full duplex, ils peuvent permettre une transmission continue.

3.3.2 Relation primaire secondaire

La ligne étant full duplex, les correspondants peuvent s'échanger en même temps de l'information. Ils peuvent donc recevoir et envoyer des données au même moment.

Pour se faire, ils possèdent tous deux un double statut fixe, comme exprimé dans le schéma ci-dessous :

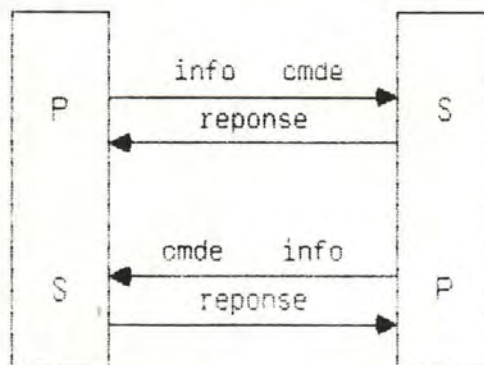


Figure 3.2: Relation primaire-secondaire

3.3.3 Procédure à fenêtre glissante

Cette procédure est basée sur l'anticipation (piggybacking).

On retient des acquittements pour les mettre dans les trames de données suivantes, dans l'autre sens.

L'importance de l'anticipation est représentée par la longueur de la fenêtre : le nombre de trames que l'on peut émettre sans acquittement reçu.

Les trames de données sont numérotées (NS) cycliquement (vu la longueur finie du champ contenant le numéro).

Lorsqu'un primaire émet, le secondaire correspondant soit bloque la fenêtre (réémission due à des erreurs de transmission, ou freinage de la communication), soit lui signifie de l'avancer par des trames de réponse, ou bien, par les numéros (NR) des trames d'informations dans l'autre sens.

Remarque : une procédure 'stop and wait' peut être considérée comme une procédure à fenêtre glissante de longueur de fenêtre égale à un.

3.3.4 Réalisation de la transparence

La technique employée ici est celle du 'bit stuffing'.

Le principe est le suivant :

lorsqu'aucune information n'est à émettre sur la ligne, la suite de bits suivante '01111110' est constamment émise.

Lorsque le niveau 2 désire émettre une trame, le niveau 1 empêche que cette suite apparaisse dans la trame.

Pour ce faire, il ajoute un '0' après chaque suite de cinq '1' de la trame à transmettre. Son correspondant, avant de livrer la trame à son niveau 2, opérera de façon inverse.

3.3.5 Types de trames

Il existe trois types de trames :

- les trames d'informations
- les trames de supervision
- les trames non numérotées.

C'est par leur champ de contrôle que l'on peut les distinguer.

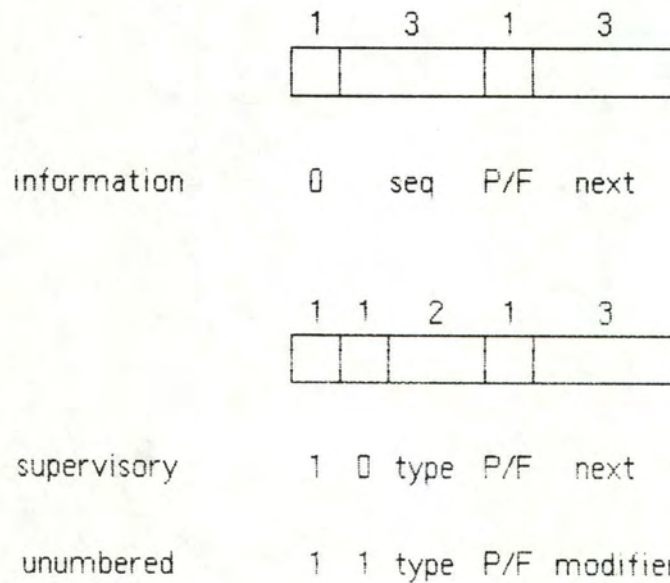


Figure 3.3: Types de champs de contrôle.

3.3.5.1 Trame d'informations

La structure de la trame d'informations est donnée à la figure suivante

| | | | | | |
|----------|---------|----------|----------|-------|----------|
| 8 | 8 | 8 | ≥ 0 | 16 | 8 |
| 01111110 | adresse | controle | donnees | check | 01111110 |

Figure 3.4: Trame d'informations

Adresse : ce champ est surtout utilisé pour les lignes multipoints, lors de l'identification du terminal.

Checksum : contient le résultat du générateur polynomial, le CRC-CCITT (Cyclic Redundancy Code).

Contrôle : ce champ contient le bit de contrôle, le numéro (NS), le P/F et le numéro (NR).

Remarque sur le P/F :

Le P/F (Poll/Final) est utilisé principalement pour transmettre des données à un groupe de terminaux.

La séquence est la suivante :

- l'ordinateur veut transmettre à un terminal : P/F = 1
- réponse du terminal : P/F = 1
- l'ordinateur transmet : n-1 trames : P/F = 0
n trame : P/F = 1.

3.3.5.2 Trames de supervision

On distingue quatre trames de supervision :

- type 0 : ACK, appelée Receive Ready.
On attend la trame suivante.
C'est utilisé lorsqu'il n'y a pas de trafic dans l'autre sens.
- type 1 : NACK, appelée Receive Not Ready.
Next contient le numéro de trame.
- type 2 : Receive Not Ready.
Acquittement de Next-1, mais pas de Next.
Employé pour temporiser.
- type 3 : Selective Reject.
Rejet d'une trame précise
(une seule a réémettre).
Utilisé dans HDLC et ADCCP.
Pas utilisé dans LAPB et SDLC.

Voici à titre d'exemple l'implémentation de ce type de trames pour LAPB :

| | 1 | 0 | type | P/F | next |
|--------|---|---|------|-----|----------|
| | | | | | |
| type 0 | 1 | 0 | 0 | 0 | P/F N(R) |
| type 1 | 1 | 0 | 0 | 1 | P/F N(R) |
| type 2 | 1 | 0 | 1 | 0 | P/F N(R) |

Figure 3.5: Trames de supervision de LAPB

3.3.5.3 Trames non numérotées

Ces trames servent au contrôle de la transmission.
Il y a, pour ce type de trames, beaucoup de différences entre HDLC et les autres protocoles.

Sont utilisés habituellement :

- DISC : disconnect.
- SNRM : Set Normal Response Mode.
Pour commencer la transmission.
Les partenaires ne sont pas égaux.
- FRMR : Frame Reject (réponse).
- UA : Unnumbered Acknowledgement (réponse).

Il existe aussi d'autres trames de contrôle pour l'initialisation, le polling, et pour échanger de l'information en la cachant à l'utilisateur (trame UI).

Sont implémentées pour le LAPB :

- DISC
- SABM : Set Asynchronous Balanced Mode.
Balanced = symétrique.
Remet les compteurs à zéro.
Les deux partenaires sont considérés comme égaux.
- SARM : Set Asynchronous Response Mode.
Equivalent de SNRM.
- UA : réponse positive.
- FRMR : Command Reject.
Equivalent de CMDR.

| | | | | | | |
|------|---|---|-------------------|---|-----|-------|
| | 1 | 1 | type P/F modifier | | | |
| | | | | | | |
| DISC | 1 | 1 | 0 | 0 | P | 0 1 0 |
| SABM | 1 | 1 | 1 | 1 | P | 1 0 0 |
| SARM | 1 | 1 | 1 | 1 | P/F | 0 0 0 |

Figure 3.6: Exemples de trames non numérotées de LAPB

4 LA COUCHE SESSION

4.1 Introduction

Ainsi que nous l'avons vu précédemment, la couche session a pour objectif de négocier les interactions entre les utilisateurs pour synchroniser les opérations effectuées sur les données.

Ce niveau est composé d'autant de modules d'échanges qu'il y a de types d'applications.

Les deux types d'applications auxquels nous nous sommes intéressés sont :

- la communication conversationnelle ou interactif
- le transfert de fichiers ou de documents.

Ces deux types d'applications sont, à notre avis, très différents.

L'échange interactif est un dialogue, tandis que l'échange de fichiers est un monologue.

Les relations entre partenaires ne sont pas les mêmes. Il s'agira pour l'échange interactif d'une relation temporaire de maître à esclave.

Pour l'échange de fichiers, on se trouvera face à une relation fixe de primaire (appelant) à secondaire (appelé), doublée lors du transfert d'une relation maître (envoyant) à esclave (recevant).

Le contrôle des actions est donc différent s'il s'agit d'une application d'échange de fichiers ou d'une application interactive.

Il est dès lors nécessaire de créer deux types de session correspondants.

Ces applications auront aussi leur propre niveau présentation correspondant.

Le modèle proposé est basé sur celui décrit par l'ISO ainsi que sur les couches supérieures du télétext.

Nous aborderons d'abord l'échange de documents. Ensuite nous verrons les différences existantes pour l'échange interactif.

La même démarche sera suivie dans le chapitre suivant traitant du niveau 6 présentation.

4.2 La session d'échange de fichiers

4.2.1 Fonctions du protocole et procédure générale

Parmi les services fournis par le niveau session, on peut distinguer principalement :

- les fonctions d'initialisation et de terminaison de l'échange
- les fonctions de transfert
- les fonctions de gestion du contrôle de l'échange
- les fonctions de gestion des erreurs.

A cet ensemble de fonctions correspondent des primitives paramétrées qui seront détaillées par la suite.

Ces primitives sont de deux types : les commandes et les réponses.

A part une exception (les commandes et réponses de transfert), à une commande correspond une réponse positive ou parfois négative, attendue dans un délais fixé (timer). Il s'agit donc d'une procédure de type "stop and wait". On verra que, pour le transfert de fichiers, au niveau document (présentation), il s'agit plutôt d'une procédure à fenêtre glissante.

On trouvera à la figure 4.1 la liste de ces primitives ainsi que leur signification.

On trouvera à la figure 4.2 un exemple de session, d'échange de ces primitives.

La procédure d'échange de ces primitives est schématisée par un graphe d'états et de transitions.

L'automate, qui sera implémenté selon le graphe, gèrera ainsi le séquençement des actions de l'appelant et de l'appelé, ainsi que les périodes d'inactivité trop importantes.

Il sera constitué en fait d'un énorme "case of" (au sens pascal) avec certaines variables d'état de la session ainsi que de différents timers.

On trouvera à la figure 4.3 le graphe d'états et de transitions.

Remarque :

dans le modèle décrit par l'ISO, la synchronisation et la resynchronisation font partie de la couche session.

La synchronisation est effectivement effectuée au niveau 5 de par le contrôle réalisé sur les actions des niveaux supérieurs.

La resynchronisation, quant à elle, nécessite des points de repère, afin de pouvoir relancer le transfert à un endroit bien déterminé.

Or, les points de repère sont fournis éventuellement (car ce n'est pas obligatoire) par le niveau 6. Il devrait y correspondre des commandes spéciales du niveau 5 afin d'envoyer ces points de repère au correspondant.

SESSION COMMANDS AND RESPONSES

| COMMAND | RESPONSE | ABBREVIATION |
|------------------------------------|--|---------------|
| SESSION ESTABLISHMENT AND CLEARING | | |
| COMMAND SESSION START | RESPONSE SESSION START POSITIVE | CSS RSSP |
| | RESPONSE SESSION START NEGATIVE | RSSN |
| COMMAND SESSION END | RESPONSE SESSION END POSITIVE | CSE RSEP |
| COMMAND SESSION ABORT | RESPONSE SESSION ABORT POSITIVE | CSA RSAP |
| INFORMATION TRANSFER | | |
| COMMAND SESSION USER INFORMATION | RESPONSE SESSION USER INFORMATION | CSUI RSUI |
| SESSION MANAGEMENT | | |
| COMMAND SESSION CHANGE CONTROL | RESPONSE SESSION CHANGE CONTROL POSITIVE | CSCC RSCCP |

Figure 4.1: Liste des primitives

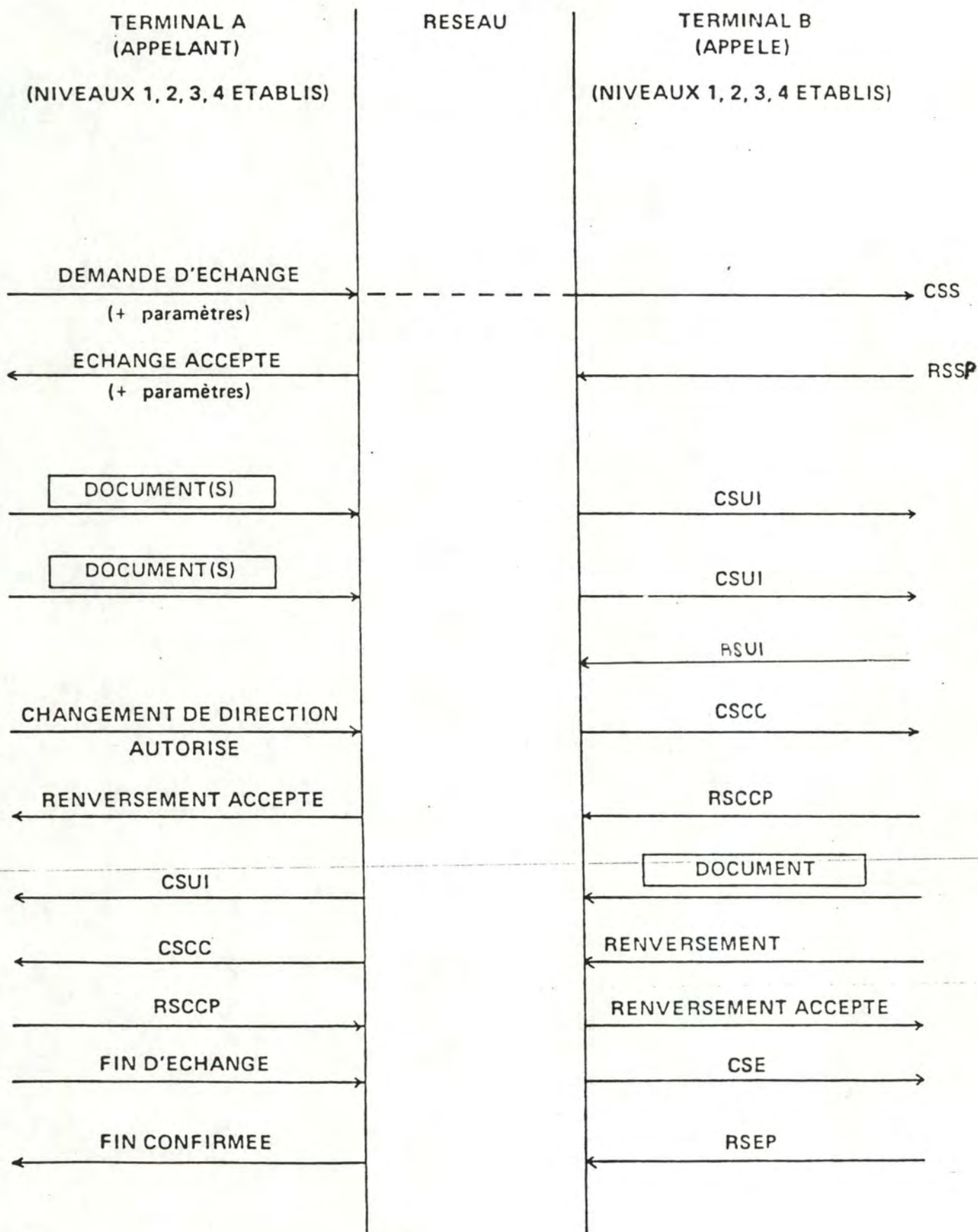


Figure 4.2: Echange

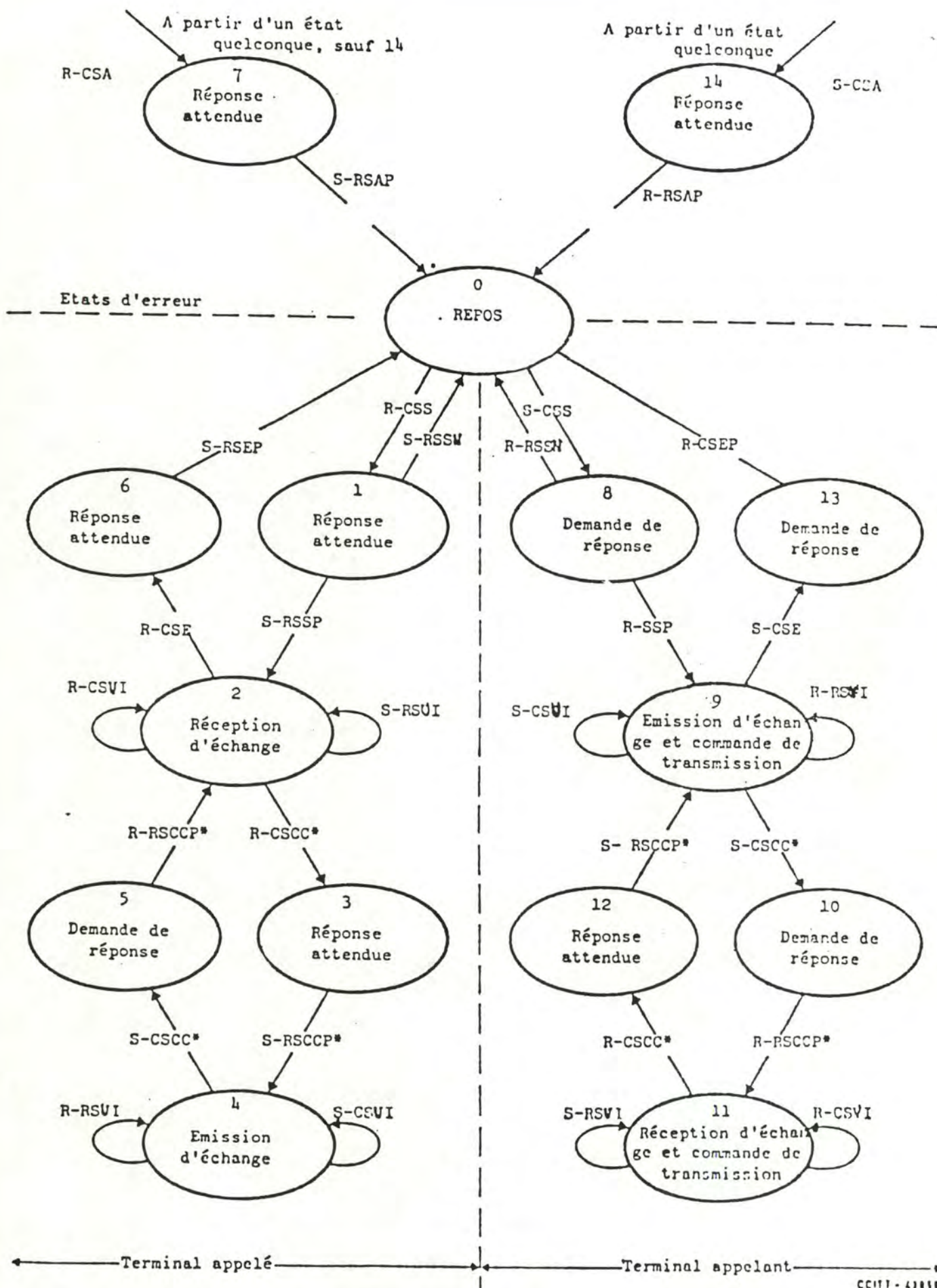


Figure 4.3: Graphe d'états et de transitions

Cette commande spéciale est inutile.

De plus, puisque le niveau 6 possède de toute façon des points de repère afin de gérer les erreurs de présentation des données, autant laisser la reprise à ce niveau, à partir du dernier point acquitté.

La resynchronisation au niveau session consistera en fait à redéfinir les relations entre partenaires afin de reprendre l'échange à partir d'états respectivement cohérents.

4.2.2 Détail des fonctions et des procédures

4.2.2.1 Initialisation de l'échange

Après la demande de connexion de l'appelant, les couches inférieures s'initialisent les unes après les autres.

Si tout se passe sans anomalie, une fois la couche inférieure adjacente initialisée avec sa réciproque, la main est donnée au niveau 5 pour l'initialisation de la session, afin d'établir les relations entre partenaires et d'arriver à un point d'accord sur l'identification de la session, les timers ainsi que les paramètres de base de présentation.

Cette initialisation se réalise grâce à la commande CSS à laquelle correspondra une acceptation RSSP ou un refus RSSN.

Un timer sera déclenché (timer BSC) pour éviter une attente infinie.

Les différents paramètres de la commande CSS sont les suivants :

- session reference :
 - terminal identifier : called et calling
 - date and time : pour l'identification
 - additional session reference number : pour identifier de façon unique
- non basic session capabilities
 - window size : fenêtre du niveau 6
 - miscellaneous session capabilities
- non basic terminal capabilities :
 - control characters set: pour le niveau 6
 - page formats : pour le niveau 6
 - miscellaneous terminal capabilities
- service identifier : interactif ou transfert (télétex)
- session service function
- timers :
 - BSC
 - abort
 - inactivité : pour le niveau 6.

Remarque sur les 'capabilities' :

- celles de base sont négociées au niveau session
- les 'extended capabilities' sont négociées au niveau 6.

Les paramètres du RSSP correspondent à ceux du CSS.

Les paramètres du RSSN sont les suivants :

- session reference
- service identifier
- session service function
- reasons

Si après cette procédure d'initialisation, les partenaires ne sont pas arrivés à un accord sur les divers paramètres, il sera mis fin à la session par un CSE.

Si par contre l'initialisation se déroule correctement, une fois la relation maître-esclave définie, la main sera donnée au niveau 6 afin qu'il s'initialise lui aussi.

4.2.2.2 Echange

Une fois la session initialisée et la relation maître-esclave établie, l'échange proprement dit peut avoir lieu. Il commencera par l'initialisation du niveau 6.

Le maître possède le contrôle de l'échange.

Son niveau 6 peut envoyer des commandes (CSUI) de données ou de contrôle.

Il recevra des réponses par RSUI du niveau 6 correspondant.

Ceci permet de vérifier si le niveau supérieur respecte bien les relations définies entre partenaires (but principal de la session).

Il est important de souligner qu'il n'y a pas de relation directe entre un CSUI et un RSUI : ce n'est pas une procédure 'stop and wait'.

Il n'est pas nécessaire d'utiliser de timer pour cette commande.

Il est à remarquer que l'inactivité, lors de l'échange de données, est gérée par le niveau supérieur à l'aide d'un timer dit d'inactivité.

Il n'y a donc pas de contrôle de données à ce niveau.

La commande CSUI ne possède pas de paramètre.

La réponse RSUI peut posséder un indicateur signalant à celui qui détient le contrôle des opérations (le maître) que l'autre (l'esclave) désirerait le prendre.

4.2.2.3 Contrôle de l'échange

On a vu précédemment que, après l'initialisation, l'appelant détenait le contrôle de l'échange.

S'il ne désire pas émettre de données mais en recevoir, la commande CSCI lui permet de passer le contrôle à l'appelé.

On a vu aussi que l'esclave pouvait demander le contrôle par un paramètre de RSUI.

La commande CSCI doit être acquittée par la réponse RSCCP.

C'est la seule réponse possible.

Son utilité est de se prévenir contre la perte du contrôle de la session.

Un timer (BSC) sera activé.

La commande CSCI et la réponse RSCCP n'ont pas de paramètre.

4.2.2.4 Fin normale de session

Une fois l'échange terminé, l'appelant, s'il possède le contrôle, demande une fin de session par la commande CSE. Il est le seul à pouvoir la demander.

Au CSE correspondra une réponse et une seule (RSEP).

Le timer BSC sera utilisé pour éviter une attente infinie.

La commande CSE possède un paramètre indiquant le maintien ou non du niveau transport (ou de la connexion). Cela est utile dans le cas où l'on désire l'établissement d'une autre session, de même type ou de type différent.

La réponse RSEP ne possède normalement pas de paramètre.

Il semblerait pourtant tout aussi utile d'offrir la possibilité à l'appelé de maintenir le transport au cas où il désirerait entreprendre une nouvelle session.

C'est pourquoi, dans le modèle que nous proposons, la réponse RSEP possèdera le même paramètre que la commande CSE.

Remarque : la commande CSE est également utilisée si la négociation des paramètres au niveau 5 échoue lors de l'initialisation.

4.2.2.5 Fin anormale de session

Comme nous l'avons cité dans le paragraphe 4.2.1, la gestion des erreurs est réalisée grâce à la commande CSA, suivie de la réponse RSAP.

Un timer spécifique est déclenché à l'envoi du CSA pour éviter à l'automate de boucler.

En effet, l'arrivée à terme du timer BSC est considérée comme une erreur.

Il s'en suit donc l'envoi de CSA avec déclenchement du timer BSC.

Dans le cas où l'appelé s'obstinerait à ne pas répondre, le timer, arrivant à son terme, redéclencherait la procédure de recouvrement d'erreurs.

L'événement suivant l'envoi d'un CSA sera soit un RSAP, soit l'arrivée à terme du timer de l'abort.

Toute autre réponse sera ignorée.

Il est à noter que seul l'appelant a le droit de déclencher ce mécanisme.

De plus, la reprise, suite à une anomalie lors de l'échange, sera effectuée par le niveau 6.

La commande CSA possède un paramètre indiquant le maintien ou non du niveau transport.

Ce maintien dépendra de la gravité de l'erreur.

La réponse RSAP ne possède pas de paramètre.

4.2.3 Gestion des erreurs au niveau 5

Lors de la session, un certain nombre d'erreurs peuvent se produire.

Elles peuvent être d'origine, de type et de gravité différents.

Le classement qui suit, est établi d'après le critère d'origine.

4.2.3.1 Erreurs du niveau 5

Nous distinguons deux catégories d'erreurs : les anomalies détectées par les timers et les erreurs non-attendues.

En ce qui concerne les erreurs de timer, on réagira de la façon suivante :

- timer BSC : séquence de fin anormale de session (abort) avec rupture de la liaison
- timer abort : rupture de la liaison.

Au niveau 6, on enregistre les paramètres et variables d'état afin de permettre la reprise (voir niveau 6).

En ce qui concerne les erreurs non-attendues : il s'agit d'erreurs du terminal local ou bien d'erreurs de procédure.

Dans ces cas, on déclenche une fin anormale de session avec maintien du transport.

Au niveau 6, on opérera une reprise à partir du dernier checkpoint, une fois qu'une nouvelle session aura été initialisée.

4.2.3.2 Erreurs au niveau 6

En ce qui concerne les erreurs du 6 entraînant un abort de la session, nous retrouvons les deux mêmes catégories qu'au point précédent,

Quant aux erreurs de timer :

une erreur peut être détectée par, soit le timer d'inactivité déclenché lors du transfert de documents, soit le timer BSC déclenché pendant les phases de contrôle du niveau 6 (hors transfert).

Dans ces cas, on enregistre les variables d'état de ce niveau et on déclenche une fin anormale de session avec rupture de la liaison transport.

En ce qui concerne les erreurs non-attendues :

il peut s'agir d'un problème de procédure qu'il soit détecté par l'automate du niveau session (exemple : demande d'envoyer une commande alors que pas le contrôle de la session), soit par l'automate du niveau 6.

Dans le cas où l'erreur est détectée au niveau 5, la procédure sera la même que celle vue au point 4.2.3.1 pour

les erreurs non-attendues.

Dans l'autre cas, on suivra la procédure ci-dessous.

On peut aussi se trouver face à des problèmes de mémoire ou de fin anormale de document due à des erreurs locales de terminal.

Dans ces cas, on enclenche une procédure spéciale au niveau 6 de gestion d'erreurs mortelles; ensuite, on entame la phase de fin anormale de session avec maintien du transport.

4.2.3.3 Autres cas

Une demande d'arrêt de session pourrait aussi être émise par l'utilisateur, le niveau application.

Nous considérons ce cas comme le plus grave :

il s'en suivra un abandon du niveau 6 et une séquence de fin anormale de session avec rupture des niveaux inférieurs.

Si une erreur grave se produit aux niveaux inférieurs, la session sera tout simplement abandonnée, et les paramètres principaux des niveaux seront enregistrés.

Toute autre anomalie survenant sera considérée comme erreur imprévue.

Il s'en suivra un enregistrement au niveau présentation, une séquence de fin anormale de session avec rupture du transport.

Lors de chaque survenance d'erreurs, un rapport devra être établi et présenté au niveau application.

4.2.4 Les timers

Nous avons vu au chapitre traitant de l'OSI que les timers étaient gérés par le moniteur.

Une couche, lorsqu'elle désire armer un timer, fait un appel au moniteur en lui spécifiant la durée et en recevant un numéro d'identification du timer.

Si le timer arrive à expiration, le moniteur le signalera à la couche intéressée qui prendra alors les mesures qui s'imposent.

Une couche doit aussi avoir la possibilité d'arrêter un timer.

Pour cela, elle fera appel au moniteur en lui spécifiant le numéro du timer à stopper.

En ce qui concerne le niveau 5, deux timers sont à employer : le timer BSC et le timer spécifique de l'abort. Le timer BSC est ainsi nommé parce qu'il est armé lors d'une procédure de type 'stop and wait'.

La nécessité d'un timer spécifique à l'abort a été expliquée plus haut.

Nous avons vu aussi que l'inactivité était gérée par le niveau 6 et que la commande CSUI et la réponse RSUI, n'ayant pas de rapport entre elles, ne nécessitaient pas de timer.

4.3 La session interactive

4.3.1 Introduction

La gestion d'un échange interactif, contrairement à l'échange de textes, se réalise presque exclusivement au niveau session.

La structure de base du schéma vu précédemment peut être reprise.

Il faut cependant y ajouter quelques mécanismes propres à l'interactif.

Les commandes, réalisant les fonctions reprises, sont :

- la commande et la réponse d'arrêt anormal de session
- la commande d'initialisation et les réponses correspondantes
- la commande de fin de session et son acquittement positif
- la commande et la réponse de transfert.

Parmi les mécanismes supplémentaires retenus, on trouve la gestion du contrôle par jeton ainsi que la suspension.

C'est en fait de par l'introduction du mécanisme de jeton qu'apparaîtront les différences les plus importantes.

Remarque :

certaines considèrent plusieurs types d'échanges interactifs : le transfert de documents (fichiers) seulement, le transfert de documents (fichiers) avec interactivité (cfr. mémoire Henrion et Waltzing), le dialogue interactif seulement.

De plus, ils considèrent deux modes possibles d'échange : le full duplex et le half duplex.

Nous considérons que :

- le transfert de document est en fait la session transfert de fichiers vu précédemment
- que le mélange des deux sessions n'est pas nécessaire et que, bien au contraire, il alourdit l'échange
- que l'interactivité seulement est le type de session interactive.

De plus, nous ne retenons que le mode half duplex, même si, a priori, il ralentit l'échange.

Notre modèle n'en est pas restrictif pour autant.

Des mécanismes permettent de passer d'un type de session à l'autre.

De plus, si le désir est émis de lancer une double communication entre les partenaires, rien n'empêche de lancer deux sessions en même temps.

4.3.2 Fonctions et mécanismes retenus

Nous reprenons ici les principaux mécanismes retenus, les caractéristiques non retenues et la justification des choix opérés.

On trouvera à la fin de cette partie une liste des différentes primitives employées ainsi que le schéma d'états et de transitions pour la session interactive (figures 4.4, 4.5 et 4.6).

4.3.2.1 Le mécanisme du jeton

La possession du contrôle de l'échange est matérialisée, pour l'échange interactif, par le jeton.

Le détenteur du jeton a le droit d'envoyer des blocs d'information, de suspendre, d'avorter ou de mettre fin à la session, de passer le jeton à son correspondant suite à une requête ou simplement parce qu'il désire recevoir de l'information.

L'esclave acquitte positivement ou négativement les commandes qu'il reçoit.

Il peut aussi demander le jeton ou la suspension.

Dans notre modèle nous avons retenu un jeton unique.

D'autres proposent quant à eux deux types de jetons : le 'terminate token' et le 'data token'.

Le 'terminate token' donne la possibilité à celui qui le détient de terminer la session;

le 'data token' confère à son propriétaire le contrôle de la session pour le reste des commandes.

Ce système à double jeton, à notre avis, complique inutilement la gestion de la session : les relations entre les deux jetons ne sont pas simples à définir, et, de plus, des tests supplémentaires sont nécessaires lors des prises de décision.

4.3.2.2 La suspension

Il peut s'avérer intéressant d'interrompre momentanément un échange interactif, par exemple, afin d'interroger brièvement une base de données (autre session interactive) ou d'entreprendre un transfert de documents.

La session sera réactivée soit après un temps convenu, soit sous demande explicite.

4.3.2.3 Fin de session

De par l'introduction du mécanisme de jeton, la fin de session interactive est assez différente de celle du transfert de fichiers.

En effet, l'appelé, s'il possède le contrôle de la transmission, peut maintenant lui aussi mettre fin soit normalement soit anormalement à la session.

De plus, en ce qui concerne la fin normale de session (CSE), une réponse négative peut être envoyée : elle permettra à l'esclave de signifier au maître qu'il désire prendre le contrôle de l'échange pour correspondre un peu plus.

4.3.2.4 Autres caractéristiques

Le flux de données interactif ne comprend pas de lien de document (points de repère).

Il n'y a donc pas de contrôle de l'échange au niveau document.

Ceci signifie que les services de synchronisation et de resynchronisation ne sont pas disponibles à l'utilisateur dans ce genre de session.

Dans certains modèles, un service de resélection est proposé.

Il s'agit de permettre à l'utilisateur en session de sélectionner une autre session avec son correspondant.

Il nous est apparu redondant avec la possibilité de terminer la session en maintenant le transport.

Ceci permet en effet de relancer une autre session sans devoir pour autant déconnecter et ensuite réinitialiser les couches inférieures.

D'autres services tels que la quarantine, l'expedited data flow, l'exception report sont aussi suggérés dans d'autres modèles.

La nécessité de ces services n'étant pas évidente, nous ne les avons pas repris.

4.3.3 Détail des fonctions et des procédures

4.3.3.1 Gestion du jeton

La demande ainsi que la passation du jeton peuvent s'opérer de plusieurs manières :

- dans les commandes et réponses attachées au transfert
- dans des messages spécifiques
- dans le refus de fin de session sans anomalie.

En ce qui concerne les SUI :

- dans RSUI : demande de jeton
- dans RSUI : passage du jeton

Remarque : un RSUI peut arriver après le passage du jeton : soit c'est une réponse du niveau document attendue avant de transiter à l'état "session receive pending token"; soit c'est une demande de jeton qui sera ignorée (il en ira de même pour tout autre message).

Quant aux messages spécifiques attachés au jeton :

- MSTP : demande de jeton
- MSTG : envoi du jeton

Dans le RSEN:

il s'agit d'un paramètre de ce message qui précise si oui ou non le correspondant, qui ne désire pas clôturer la session, désire le jeton.

Le jeton sera envoyé par un des mécanismes vu précédemment.

Remarques :

- on pourrait concevoir les messages spécifiques de passage de jeton comme des commandes nécessitant des réponses, ceci pour se prévenir de la perte ou du dédoublement du jeton.
Ce mécanisme n'est pas nécessaire ainsi que nous le verrons plus loin.
- le mécanisme de passage de contrôle (CSCC, RSCCP), utilisé lors d'une session transfert de fichiers, n'est plus de mise ici.

4.3.3.2 Gestion de la suspension

Nous avons vu plus haut que seul le détenteur du jeton avait le droit de suspendre la session.

Toutefois, une requête de suspension peut provenir de son correspondant soit par le message CSUR (command session suspend request), soit par un paramètre de RSUI.

Suite à cette demande (CSUR ou RSUI), soit la suspension sera réalisée par le détenteur du jeton qui désire dans ce cas lui aussi suspendre la session; soit, ne désirant pas entreprendre de lui-même la suspension, le détenteur du jeton passe le contrôle au demandeur qui

| PRIMITIVE | ABBREVIATION |
|--------------------------------------|--------------|
| Session establishment and clearing | |
| Command Session Start | CSS |
| Response Session Start Positive | RSSP |
| Response Session Start Negative | RSSN |
| Command Session End | CSE |
| Response Session End Positive | RSEP |
| Response Session End Negative | RSEN |
| Command Session Abort | CSA |
| Response Session Abort Positive | RSAP |
| Information transfer | |
| Command Session User Information | CSUI |
| Response Session User Information | RSUI |
| Session management | |
| Message Session Token Please | MSTP |
| Message Session Token Give | MSTG |
| Command Session Suspend | CSSU |
| Response Session Suspend Positive | RSSUP |
| Response Session Suspend Negative | RSSUN |
| Command Session Suspend Request | CSUR |
| Command Session Reactivate | CSR |
| Response Session Reactivate Positive | RSRP |
| Response Session Reactivate Negative | RSRN |

Figure 4.4: Liste des primitives

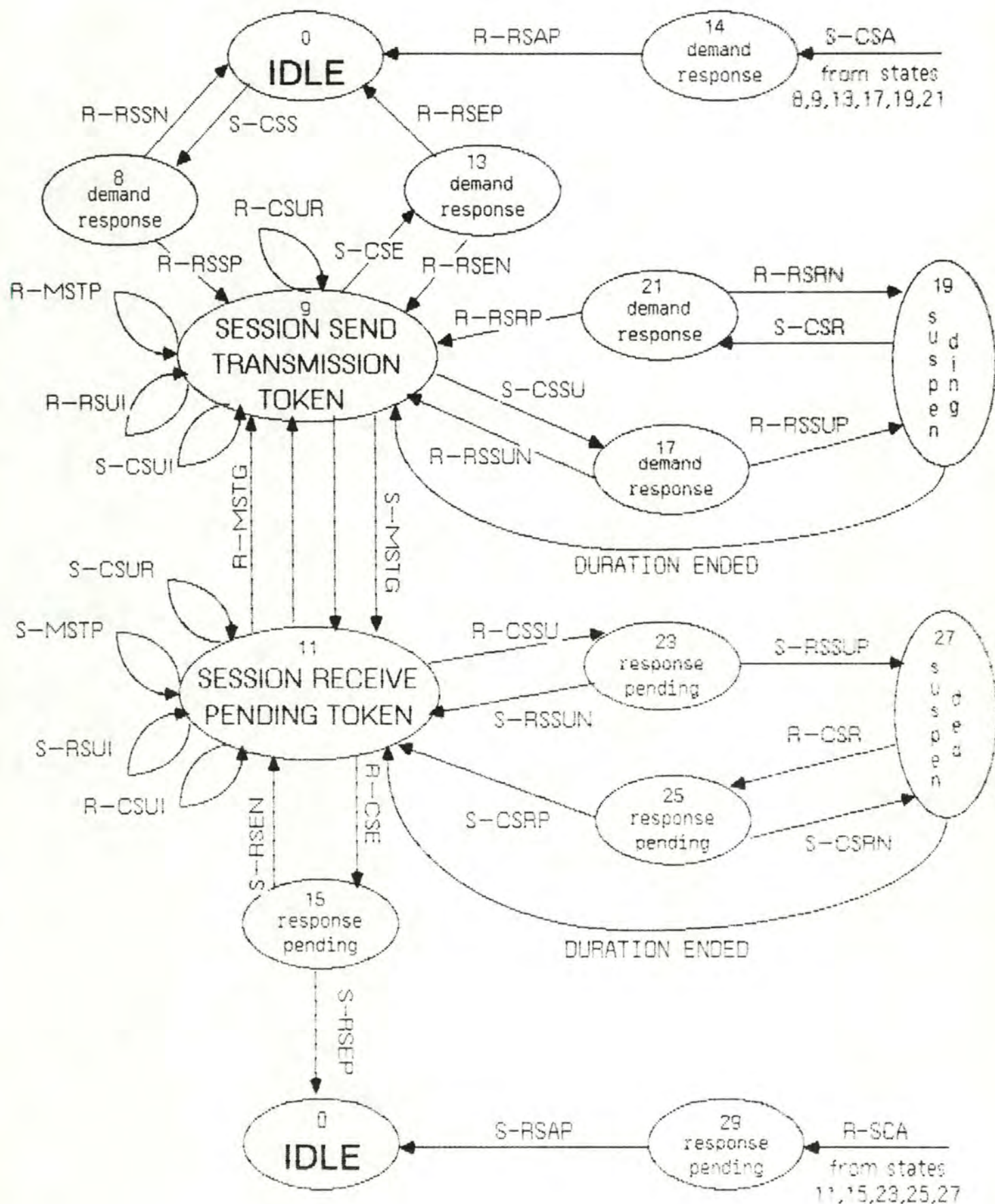


Figure 4.5: Schéma d'états et de transitions (appelant)

alors réalisera la suspension.

La procédure de suspension est la suivante :

- commande CSSU
- réponse :
 - positive (RSSUP) : la session est suspendue pour un temps déterminé
 - négative (RSSUN) : on revient à l'état précédent.

Ce mécanisme nécessite donc un timer BSC ainsi qu'un timer de durée de suspension.

Le détenteur du jeton peut réactiver la session avant le temps désiré de la manière suivante :

- commande de réactivation : CSR
- réponse :
 - positive (RSRP) : on revient à la communication (on arrête le timer de suspension)
 - négative (RSRN) : on revient à l'état suspendu.

Si le timer de suspension échoue à son terme, un réveil du niveau supérieur sera opéré et la communication reprendra son cours.

4.3.3.3 Procédure de fin de session

La procédure de fin normale de session peut être déclenchée par le détenteur du jeton.

Le mécanisme est le suivant :

- CSE : demande de fin avec maintien ou non du transport
- réponse :
 - RSEP : acquittement positif avec ou sans maintien du transport
 - RSEN : acquittement négatif avec paramètre indiquant une demande de jeton.

Le timer BSC sera donc enclenché lors du CSE.

Remarques :

si le paramètre du RSEP demande le maintien du transport, cela signifie que l'esclave désire une session différente.

le mécanisme RSEN est ici possible car la relation entre les partenaires est une relation de maître à esclave.

Par contre, pour le transfert de fichiers, la relation initiale est de primaire (appelant) à secondaire (appelé), doublée, lors du transfert, par une relation maître (envoyant) à esclave (recevant).

Si le secondaire désire prendre le contrôle réel du transfert, l'appelant doit d'abord clore la session pour lui permettre d'ouvrir une nouvelle session par la suite.

4.3.3.4 Initialisation

La procédure d'initialisation de la session interactive est identique à celle du transfert de fichiers.

Il existe toutefois quelques différences quant aux paramètres.

En effet, si l'on se base sur le protocole télétext, toute négociation de paramètre s'effectue au niveau 5. Nous garderons pourtant la possibilité de négocier certains paramètres extraordinaires au niveau 6.

Les paramètres retenus sont :

- session reference
- non basic terminal capability
- timers
- service identifier
- service function

Les paramètres non repris sont le timer d'inactivité et le non basic session capability.

4.3.3.5 Echange interactif

Comme pour le transfert de fichiers, l'échange interactif est réalisé à l'aide de la commande CSUI et de la réponse RSUI.

Nous rappelons qu'il n'y a pas de relation entre les deux messages.

Rappelons aussi que le CSUI possède ici deux paramètres, l'un servant à la gestion du jeton, l'autre à la gestion de la suspension.

Il en va de même pour la réponse RSUI.

Il est important de noter que l'échange interactif n'est supporté par aucun contrôle de niveau 6 (séquençement, inactivité).

4.3.4 Gestion des erreurs

4.3.4.1 Erreurs générales

Toute erreur déclenchera ici le mécanisme de fin anormale de session (procédure abort).

Cette procédure (abort) est identique à celle de la session transfert de fichiers avec en plus la possibilité laissée à l'appelé de déclencher lui aussi le mécanisme s'il possède le jeton.

Dans le cas des erreurs de timers de type BSC ou abort, la session sera avortée avec rupture de transport. Pour les autres erreurs de protocole, le transport sera maintenu.

L'utilisateur (niveau application) peut lui aussi demander une fin anormale de session, par exemple, s'il remarque une inactivité trop importante de la part de son partenaire ou suite aux erreurs de niveau 6.

4.3.4.2 Erreurs portant sur le jeton

Pour dégager les anomalies pouvant survenir lors de la gestion du jeton réglant une session interactive, nous nous sommes basés sur celles qui ont été mises en évidence dans la méthode d'accès par jeton pour un réseau local en anneau.

Dans un anneau, la station possédant le jeton est la seule à pouvoir envoyer des données sur le réseau. Lorsqu'elle a fini, elle passe le jeton à une autre station qui veut émettre.

Les anomalies qui peuvent se produire sur le jeton sont classées en trois catégories :

- la perte du jeton
- le dédoublement du jeton
- le jeton bloqué sur l'état occupé.

Ces erreurs proviennent de perturbations électriques sur le réseau qui modifient la configuration binaire du jeton.

Les erreurs de gestion dans une session interactive ne peuvent provenir de perturbations sur le réseau car elles sont récupérées aux niveaux inférieurs.

Voyons néanmoins si ces anomalies peuvent se produire lors d'un échange interactif pour des raisons différentes.

4.3.4.2.1 La perte du jeton

Le jeton, transmis par MSTG ou dans un paramètre de CSUT ou de RSEN, peut être perdu par un mauvais encodage.

Dans ce cas, le côté appelant, ayant envoyé le jeton, passe de l'état 9 (session send and transmission token) à l'état 11 (session receive pending token) alors que le côté appelé qui devrait passer à l'état 4 reste à l'état 2 car il n'a pas reçu le jeton.

Dans ces états, les côtés appelant et appelé ne peuvent qu'envoyer une demande de jeton ou de suspension de la session.

Or, l'état 11, pour le côté appelant, n'est susceptible que de recevoir des messages de l'état 4 du côté appelé, et pas ceux de l'état 2.

Il y aura donc une arrivée de message non prévu, ce qui conduira à un erreur de procédure.

Dans ces deux états, il se peut qu'aucun des côtés n'envoie de messages car leur fonction principale est d'en recevoir et de les acquitter.

L'utilisateur s'apercevra alors de l'inactivité et demandera soit le jeton, soit une suspension de la session, ce qui conduira aussi à une erreur de procédure.

4.3.4.2.2 Le jeton dupliqué

Suite à une erreur d'encodage des paramètres spécifiant l'envoi du jeton, les deux côtés possèderaient le jeton.

Le côté appelant restant à l'état 9 et le côté appelé passant à l'état 4, ils se trouvent tous les deux en mode transmission et donc chaque côté enverra des commandes et en recevra, alors que ces états ne sont pas prévus pour recevoir des commandes.

Il en résultera donc également une erreur de procédure.

4.3.4.2.3 Jeton bloqué sur l'état occupé

Dans le cas d'une session interactive, une variable d'état indique dans quel état se trouve l'automate, s'il possède ou non le jeton.

Si à cause d'une erreur cette variable change alors qu'aucun événement pouvant causer ce changement n'est survenu, les côtés appelant et appelé se trouveront dans des états non compatibles, ce qui nous ramène au deux cas vus précédemment.

Il y aura donc une erreur de procédure

Ce cas n'est pas équivalent à celui de l'anneau utilisant la méthode d'accès par jeton.

En effet, dans cette méthode, le jeton peut circuler sur l'anneau à l'état libre : aucune station ne le possède.

Dans une session interactive, au moins un des deux partenaires possède le jeton à tout moment.

4.3.4.2.4 Conséquence d'une erreur de procédure

Une erreur de procédure nécessite une réinitialisation de la session afin de se retrouver dans des états cohérents.

On réinitialisera la session avec les derniers paramètres utilisés aux niveaux 5 et 6.

Le transport sera maintenu.

4.3.5 Les timers

La procédure d'utilisation est la même que celle exposée pour l'échange de fichiers.

Rappelons toutefois les timers utilisés pour l'échange interactif :

- timer BSC
- timer de suspension
- timer de l'abort.

5 LA COUCHE PRESENTATION

5.1 Introduction

Ainsi que nous l'avons vu dans le chapitre traitant de l'OSI, la couche présentation, encore appelée niveau document, a pour objectif de pouvoir interpréter la signification des données échangées entre utilisateurs.

Nous avons vu aussi au chapitre intitulé couche session qu'à chaque type d'applications correspondait un module présentation.

Nous aborderons en premier lieu le niveau présentation pour l'échange de fichiers (documents).

Ensuite, nous examinerons les différences existantes pour l'échange interactif.

5.2 Echange de fichiers

5.2.1 Fonctions du protocole et procédure générale

La fonction principale du niveau présentation est la préparation sous une forme adéquate du fichier, soit pour l'envoyer, soit pour le présenter au niveau supérieur.

Le fichier sera donc codé, soit :

- en forme standard acceptée par les deux parties (fichier virtuel)
- directement à l'envoi en une forme adaptée pour la réception
- en forme de présentation optimale suite à la comparaison des possibilités des deux partenaires.

Ce n'est que dans le cas d'un système de fichier virtuel que le niveau 6 recevant devra entamer une procédure de transformation du fichier.

Les paramètres de présentation, confirmant ceux échangés à l'initialisation de la session, seront discutés entre partenaires à l'initialisation du niveau 6.

La deuxième fonction principale, dérivant de la première, est la récupération d'erreurs de présentation lors du transfert.

Nous en avons déjà parlé au chapitre traitant de la session de transfert de fichiers.

Cette fonction nécessite le découpage du fichier en pages numérotées (identifiées) par des points de repère.

Ce mécanisme appelé aussi reprise ou resynchronisation sera détaillé dans un paragraphe suivant (relancement du transfert).

On trouvera, dans les paragraphes traitant du transfert, le mécanisme de fenêtre glissante à numérotation non cyclique.

Remarques :

- dans notre modèle, nous proposons que le codage du document soit réalisé par page, afin d'éviter tout travail inutile en cas d'erreur entraînant une fin anormale des niveaux 6, 5 et 4
- nous ne traiterons pas le codage.

Outre ces fonctions principales, on retrouve les fonctions habituelles suivantes :

- initialisation de la présentation
- phase de transfert du document (lancement, transfert, fin)
- gestion des erreurs.

A ces fonctions correspondent un ensemble de primitives paramétrées qui seront détaillées par la suite.

On trouvera à la figure 5.1 une liste de ces primitives ainsi que leur signification.

On trouvera à la figure 5.2 un exemple d'enchaînement de ces primitives.
On trouvera aux figures 5.3 et 5.4 le graphe d'états et de transitions.

5.2.2 Détail des fonctions et des procédures

5.2.2.1 Initialisation

L'initialisation est réalisée par le maître grâce à la commande CDCL à laquelle correspond la réponse RDCLP.
Un timer BSC sera activé.

Les valeurs des paramètres de présentation compris dans le CDCL, sont soit :

- la confirmation des paramètres de présentation échangés lors de l'initialisation de la session
- des valeurs spéciales, liées à l'application, que le terminal esclave pourrait avoir besoin (à titre de renseignement).

Cette commande sert aussi à se renseigner sur les possibilités de stockage de l'esclave.
Le timer d'inactivité (voir plus loin) pourra être renégocier.

Les paramètres du CDCL sont les suivants :

- inactivity timer
- storage capacity negotiation
- non basic terminal capabilities :
 - graphic character sets
 - control characters
 - page format
 - miscellaneous terminal capabilities.

La réponse RDCLP sert à confirmer ou à montrer son désaccord sur les paramètres du CDCL.

Elle contient une liste des capacités extraordinaires que le terminal possède (à titre de renseignement).

En ce qui concerne la capacité mémoire, on peut indiquer la confirmation que la mémoire désirée est disponible ou non, la quantité totale disponible, ...

La réponse RDCLP contient un paramètre de plus que le CDCL, à savoir :

- acceptance of CDCL parameters.

| COMMAND | RESPONSE | ABBREVIATION |
|----------------------------------|--|--------------|
| DOCUMENT CONTROL | | |
| COMMAND DOCUMENT START | | CDS |
| COMMAND DOCUMENT CONTINUE | | CDC |
| COMMAND DOCUMENT CAPABILITY LIST | | CDCL |
| | RESPONSE DOCUMENT CAPABILITY LIST POSITIVE | RDCLP |
| COMMAND DOCUMENT END | | CDE |
| | RESPONSE DOCUMENT END POSITIVE | RDEP |
| COMMAND DOCUMENT DISCARD | | CDD |
| | RESPONSE DOCUMENT DISCARD POSITIVE | RDDP |
| COMMAND DOCUMENT RESYNCHRONIZE | | CDR |
| | RESPONSE DOCUMENT RESYNCHRONIZE POSITIVE | RDRP |
| INFORMATION TRANSFER | | |
| COMMAND DOCUMENT INFORMATION | | CDUI |
| ERROR RECOVERY | | |
| | RESPONSE DOCUMENT GENERAL REJECT | RDGR |
| COMMAND DOCUMENT PAGE BOUNDARY | | CDPE |
| | RESPONSE DOCUMENT PAGE BOUNDARY POSITIVE | RDPBP |
| | RESPONSE DOCUMENT PAGE BOUNDARY NEGATIVE | RDPBN |

Figure 5.1: Liste des primitives

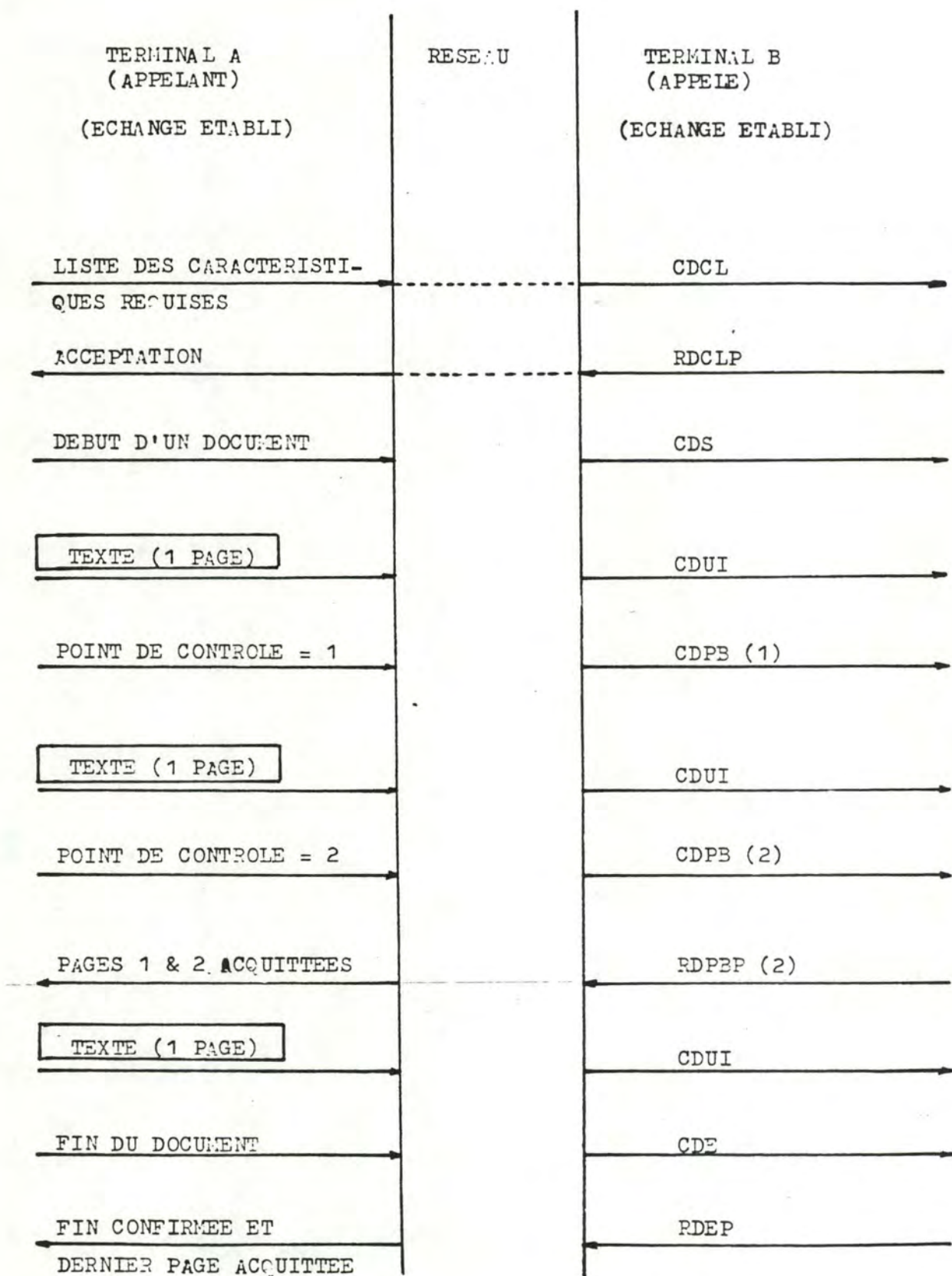


Figure 5.2: Exemple d'enchaînement

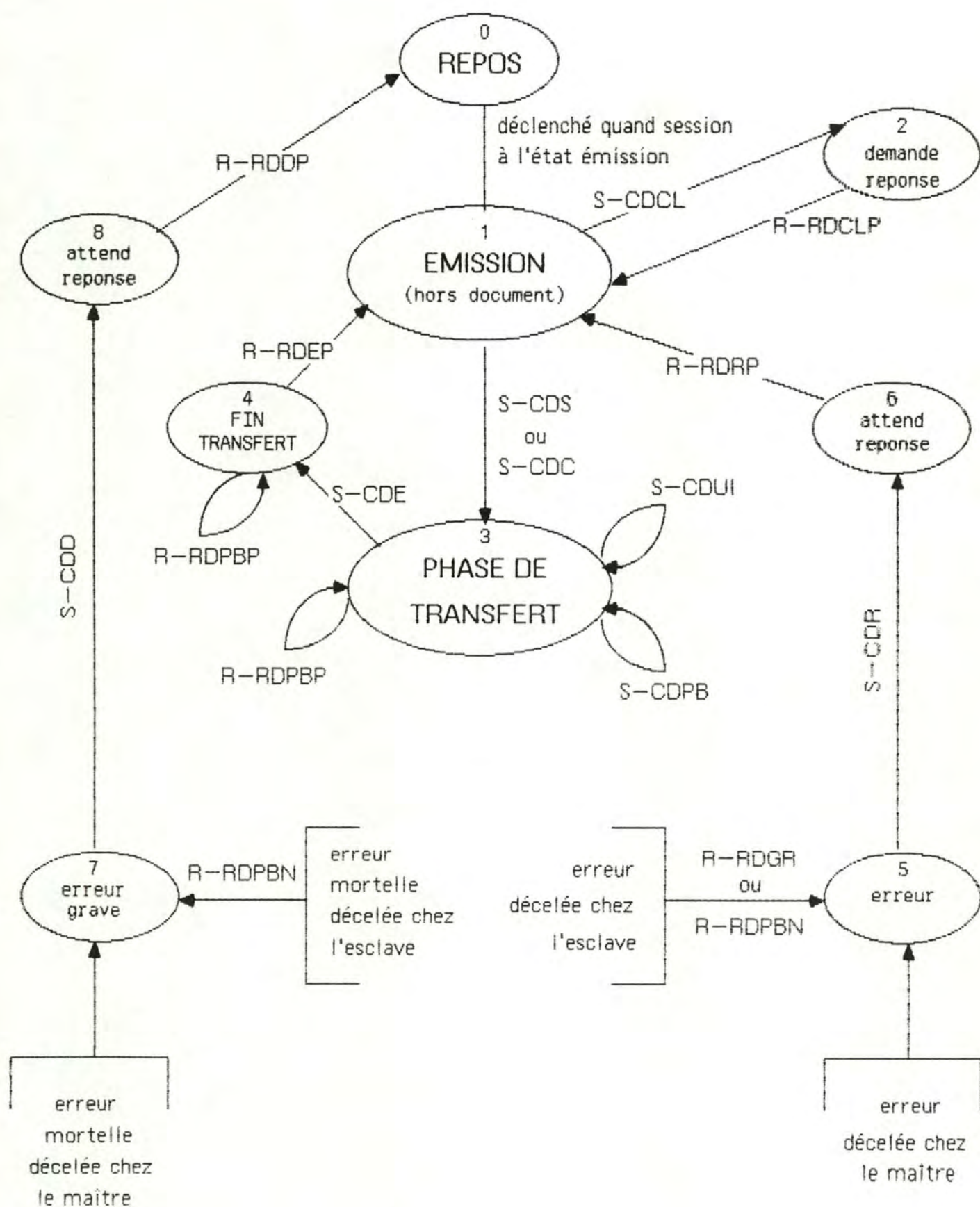


Figure 5.3: Graphe d'états et de transitions (maître)

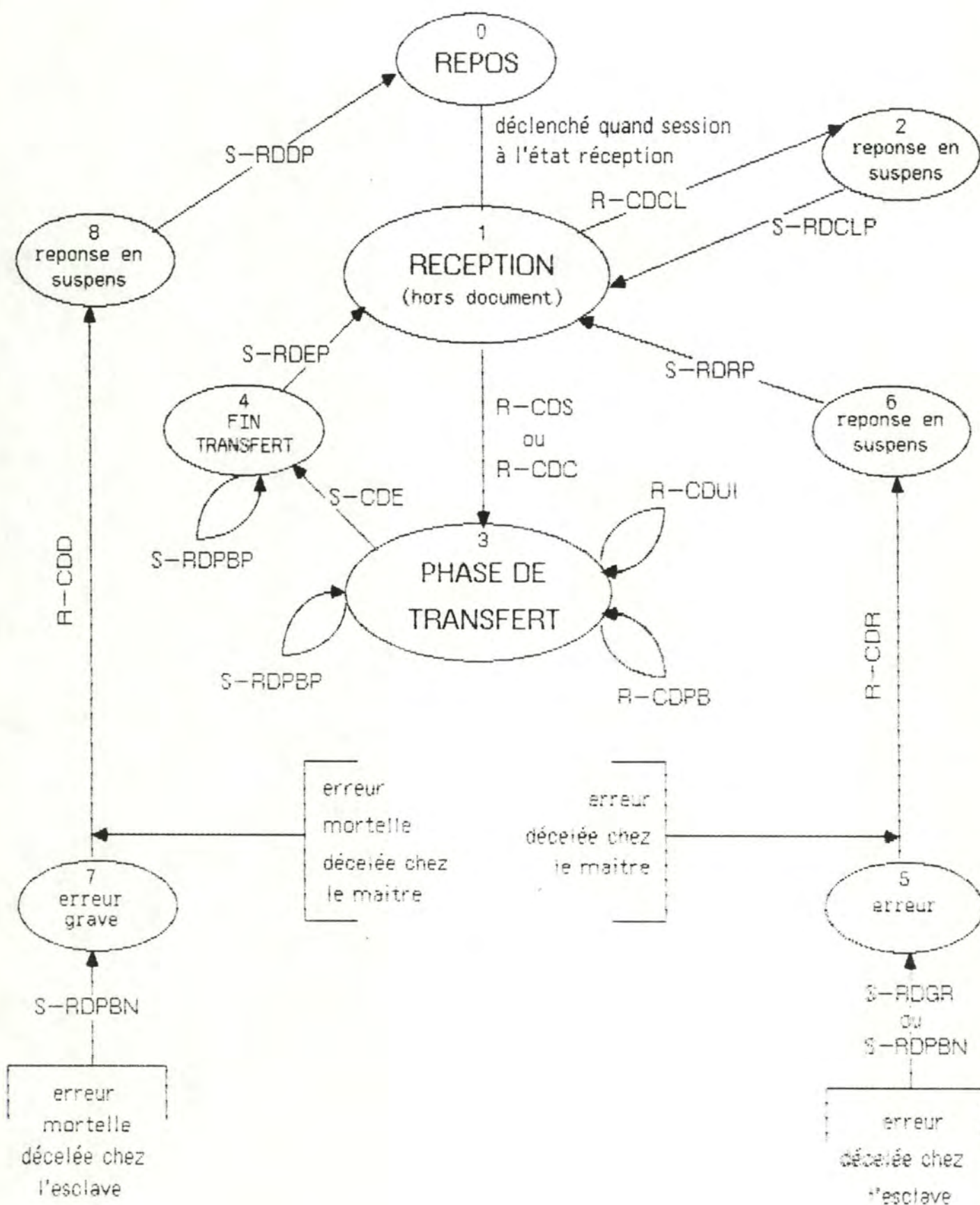


Figure 5.4: Graphe d'états et de transitions (esclave)

5.2.2.2 Lancement et relancement de l'échange

Le lancement et le relancement marque le début de la phase de transfert.

Il existe un timer réglant l'échange à ce niveau afin d'éviter l'attente infinie d'une réponse du correspondant. Il s'agit du timer d'inactivité.

Il sera déclenché lors du lancement ou du relancement du transfert et réinitialisé lors d'une réponse correcte des points de vue syntaxique et de procédure, survenant lors de cette phase de transfert.

L'arrivée à terme de ce timer sera considérée comme une erreur mortelle.

Rappelons que la phase de transfert est composée du lancement ou du relancement, du transfert et de la fin normale de document.

La commande CDS indique le commencement d'un document et de la phase de transfert.

Elle est utilisée après l'initialisation (CDCL).

Il n'y correspond aucune réponse sauf en cas d'erreur (RDGR).

Les paramètres de CDS sont les suivants :

- document reference number : un document porte un numéro d'identification dans le cas où la session comporterait plusieurs transferts
- service interworking identifier : si les deux services en présence ne sont pas de même type (exemple : télex-télétext)
- document type identifier : normal, d'opérateur, de contrôle (télétext)
- non-basic terminal capabilities : au cas où le document doit utiliser des qualités standard optionnelles (en plus que celles définies lors de l'initialisation de la session).

Le transfert d'un document peut être interrompu.

La commande CDC sert à le relance de l'échange (synonymes : reprise, resynchronisation).

Elle n'attend pas de réponse, sauf en cas d'erreur (RDGR).

Elle possède les mêmes paramètres que CDS avec en plus :

- checkpoint reference number : dernier point de repère acquitté à partir duquel l'échange va reprendre
- session reference : si la session avait dû être interrompue (télétext)

5.2.2.3 Transfert

5.2.2.3.1 Mécanisme de la fenêtre

Le transfert des données, à ce niveau, est réglé par une procédure à fenêtre glissante à numérotation non cyclique.

On a choisi ce mécanisme afin de permettre une transmission continue.

Cette procédure permet aussi la récupération d'erreurs de document ainsi qu'un contrôle de flux (normalement réalisé par le niveau 4).

Le fichier à transférer est découpé en pages.

Chaque page, dans le mode de base (voir performance), est identifiée par un point de repère envoyé par une commande spéciale (CDPB) après l'envoi de la page (CDUI).

La taille de la fenêtre a été définie suite à l'initialisation de la session.

Le maître peut envoyer un nombre de pages égal à la largeur de la fenêtre.

L'esclave, recevant les pages numérotées à partir de 1, acquitte positivement (RDPBP) ou négativement (RDPBN).

L'acquiescement positif permet au maître d'avancer sa fenêtre d'émission.

A titre d'exemple :

fenêtre (4,5,6)

R-RDPBP 5

fenêtre (6,7,8).

La réception d'un RDPBN (pour erreur de document) impliquera une réémission de pages.

Ce mécanisme permet de récupérer les erreurs de présentation (réémission de la mauvaise page), de contrôler le flux (l'envoi successif de RDPBN empêche le maître de faire progresser sa fenêtre d'émission) et de contrôler le séquençement.

5.2.2.3.2 Remarque sur le mécanisme de la fenêtre

La procédure réelle de transfert dépendra des accords convenus préalablement.

Etant donné une fenêtre de taille N, on aura :

- à l'émission :
 - soit : $N ((CDUI) + (CDPB))$
 - soit : $N ((M CDUI) + (CDPB))$
avec M le nombre de pages que l'on peut émettre avant un checkpoint
- à la réception :
 - RDPBN : à tout moment (pour erreur de document)
=> réémission de l'ensemble des pages de la fenêtre

- RDPBP : à tout moment
 - soit on impose un RDPBP par CDPB
 - soit on accepte un RDPBP avec un numéro quelconque compris dans la fenêtre.

Ces conventions sont complémentaires au schéma d'états et de transitions.
Elles doivent être obligatoirement définies.

5.2.2.3.3 Détail des commandes

La commande CDUI d'envoi de page ne possède pas de paramètre.

La commande CDPB d'envoi de point de repère possède un paramètre :

- checkpoint reference number : point de référence de la dernière page.

La réponse RDPBP possède deux paramètres :

- checkpoint reference number : numéro de page acquittée
- receiving ability jeopardized : possibilité ou non de continuer à recevoir le trafic.

La réponse RDPBN possède un paramètre :

- reasons : raison du refus (plusieurs degrés de gravité).

5.2.2.3.4 Performance

Vu le désir de transmission continue qui implique une anticipation à l'émission et la possibilité de recevoir des réponses en même temps que d'émettre, le mode full duplex serait préférable.

La performance du protocole peut être améliorée des façons suivantes :

- si peu d'erreurs à l'émission :
 - augmentation de la longueur de la page
 - augmentation de la longueur de la fenêtre
 - réduction de la fréquence des points de repère
- à la réémission :
 - réémettre uniquement la page erronée (paramètre supplémentaire)
 - et non pas à partir de la page erronée (toute la fenêtre - les pages acquittées).

5.2.2.4 Fin normale de document

La commande CDE est utilisée pour indiquer la fin du transfert de document.

Elle contient le dernier point de repère auquel une réponse devra être faite.

Suite à cette commande, peuvent arriver des réponses RDPBP de points de repère précédents.

Une fois que le recevant aura examiné la dernière page, il répondra soit par un RDEP (acquiescement positif), soit par un RDPBN (en cas d'erreur de document).

La réponse RDEP contiendra le même point de repère que le CDE.

Il est à noter que nous sommes toujours dans la phase de transfert.

Le problème d'attente de réponse est donc géré par le timer d'inactivité.

5.2.2.5 Fin anormale de document

Une erreur de document peut être constatée par le maître ou l'esclave.

Il s'agit soit d'une erreur de séquençement, soit d'une erreur de codage de présentation.

Les réactions seront les suivantes :

- par le maître : procédure CDR
- par l'esclave :
 - envoi de RDPBN
 - procédure CDR

La procédure CDR est la suivante :

- commande : CDR (indique à l'esclave la resynchronisation)
- réponse :
 - RDRP : confirmation
 - RDGR : erreur à la réception du CDR.

La commande CDR peut contenir un paramètre indiquant l'erreur survenue.

La réponse RDRP ne contient pas de paramètre.

Un timer BSC sera déclenché lors de l'envoi de CDR.

Remarque :

il existe d'autres types d'erreurs;

ils sont détaillés dans la gestion des erreurs.

5.2.3 Gestion des erreurs

Nous distinguons deux grandes catégories d'anomalies : les anomalies détectées par l'arrivée à terme d'un timer, les autres erreurs de protocole.

5.2.3.1 Erreurs de timer

Il existe deux types de timers.

Le premier type comprend les timers déclenchés lors de l'envoi d'un message devant être acquitté.

Il s'agit des timers BSC.

Le second type concerne le timer d'inactivité déclenché lors de la phase de transfert.

L'expiration de ces timers provoquera un enregistrement des paramètres de présentation (afin de permettre la reprise), un abort de la session avec rupture du transport (comme faire 4.2.3.2).

On retiendra la dernière page acquittée.

5.2.3.2 Autres erreurs de protocole

Il existe plusieurs types d'erreurs dont les conséquences sont de gravité différente :

- les erreurs non mortelles, suite auxquelles on peut reprendre l'échange sans mettre fin à la session
- les erreurs mortelles, occasionnant un abort de la session (cfr 4.2.3.2).

Les erreurs de procédure sont considérées comme mortelles, qu'elles soient détectées au niveau 5 (traitées alors à ce niveau, cfr 4.2.3.2 et 4.3.2.1) ou au niveau 6. Sont aussi mortelles, les problèmes de mémoire, de fin anormale de document dues à des erreurs locales de terminal (cfr 4.3.2.2).

Les erreurs non mortelles sont en fait des pseudo-erreurs, des erreurs prévues : réponse négative de transfert, refus d'un paramètre, commande polluée.

La gestion de ces deux types d'erreurs est très différente.

Nous verrons d'abord les erreurs mortelles, ensuite, les erreurs non mortelles.

5.2.3.2.1 Gestion des erreurs mortelles

Une erreur mortelle se produit soit :

- chez l'esclave : RDPBN (avec les causes)
- chez le maître : erreur propre ou réception d'un RDPBN.

Il ne s'agit pas du RDPBN d'erreur de document.

La procédure est alors la suivante :

- commande : CDD
- réponse : RDDP.

On retiendra la dernière page avant le dernier CDC.
Les autres seront abandonnées.

On lancera ensuite un abort session avec maintien du transport.

Ce type d'erreur peut donc occasionner plus de perte de pages transmises que l'erreur de timer.

La commande CDD contient un paramètre indiquant l'erreur survenue.

La réponse RDDP n'a pas de paramètre.

Un timer BSC sera déclenché lors de l'envoi de CDD.

5.2.3.2.2 Gestion des erreurs non mortelles

Une erreur non mortelle se produit soit :

- chez l'esclave :
 - accepte pas une commande hors du transfert (CDC, CDS, CDLC, CDR)
réaction : RDGR
 - accepte pas une commande de transfert (erreur de document : mauvais séquençement, mauvaise page)
réaction : RDPBN
- chez le maître :
 - erreur de document
 - réception d'un message vu plus haut

La procédure suivie par le maître est alors la procédure de fin anormale de document, avec ou non, une renégociation des paramètres (CDCL, RDCLP).

La reprise pourra se faire grâce à CDC.

La réponse RDGR contiendra la cause du refus (paramètres non acceptés, commande non comprise) dans un paramètre qui lui est attaché.

5.2.4 Les timers

La procédure d'utilisation des timers est la même que celle vue dans le chapitre traitant de la session.

Nous rappelons les timers utilisés par le niveau 6 transfert de fichiers :

- timer BSC (hors phase de transfert)
- timer d'inactivité (phase de transfert).

5.3 Interactif

5.3.1 Introduction

Le niveau présentation pour un échange interactif est un niveau assez réduit.

La fonction principale est de présenter de façon adéquate les données au terminal.

Il existe plusieurs classes de terminaux dont la puissance, exprimée par le nombre et la qualité des fonctions que le terminal offre, varie.

Le but est de trouver un compromis entre les classes en présence.

Les données seront présentées soit :

- en une forme standard
- directement en une forme adaptée pour la réception
- en une forme optimale suite à la comparaison des possibilités des deux partenaires.

La deuxième fonction, la récupération d'erreurs de document (séquencement, présentation), n'est pas reprise pour l'interactif.

A part les contrôles opérés au niveau session, tout contrôle est laissé à l'utilisateur, au niveau application. On ne retrouvera donc pas le mécanisme des points de repère, de la fenêtre glissante, du timer d'inactivité.

Il n'y a pas non plus de gestion générale d'erreurs à ce niveau, excepté pour l'initialisation.

Toute erreur sera signalée au niveau supérieur qui prendra les mesures qui s'imposent (abort ou continuation).

Parmi les fonctions habituelles, on retrouve :

- le transfert de données (échange interactif)
- l'initialisation du niveau.

On trouvera à la figure 5.5 la liste des primitives paramétrées correspondant à ces fonctions.

On trouvera à la figure 5.6 le graphe d'états et de transitions pour un échange interactif.

5.3.2 Détail des fonctions et des primitives

5.3.2.1 Initialisation du niveau

Le modèle du télétext ne reprend pas l'initialisation du niveau 6 pour l'échange interactif.

La négociation des capacités extraordinaires des terminaux, pour ce modèle, est donc laissée au niveau supérieur.

Afin de rester cohérent avec ce que nous avons dit pour l'échange de fichiers, nous reprendrons cette possibilité de négociation à ce niveau.

Le mécanisme d'initialisation est donc semblable à celui vu pour l'échange de fichiers.

L'initialisation est réalisée par le maître grâce à la commande CDCL à laquelle correspond la réponse RDCLP.

Les valeurs des paramètres de présentation compris dans le CDCL, sont soit :

- la confirmation des paramètres de présentation échangés lors de l'initialisation de la session
- des valeurs spéciales, liées à l'application, que le terminal esclave pourrait avoir besoin (à titre de renseignement).

Les paramètres du CDCL sont les suivants :

- non basic terminal capabilities
 - graphic character sets
 - control characters
 - page format
 - miscellaneous terminal capabilities

La commande RDCLP sert à confirmer ou à montrer son désaccord sur les paramètres du CDCL.

Elle contient une liste des capacités extraordinaires que le terminal possède (à titre de renseignement).

La réponse RDCLP contient un paramètre de plus que le CDCL, à savoir :

- acceptance of CDCL parameters.

Il y a un contrôle d'erreurs sur cette commande, comme pour l'échange de fichiers.

Un timer BSC sera déclenché.

L'arrivée à terme de ce timer sera signalée au niveau supérieur qui avisera.

Si l'esclave ne comprend pas la commande, il répondra par RDGR.

Ce message sera signalé au niveau supérieur qui avisera.

| PRIMITIVE | ABBREVIATION |
|---|--------------|
| Document initialisation | |
| Command Document Capability List | CDCL |
| Response Document Capability List Positive | RDCLP |
| Response Document General Reject | RDGR |
| Information transfer | |
| Command Document Information | CDUI |

Figure 5.5: Liste des primitives

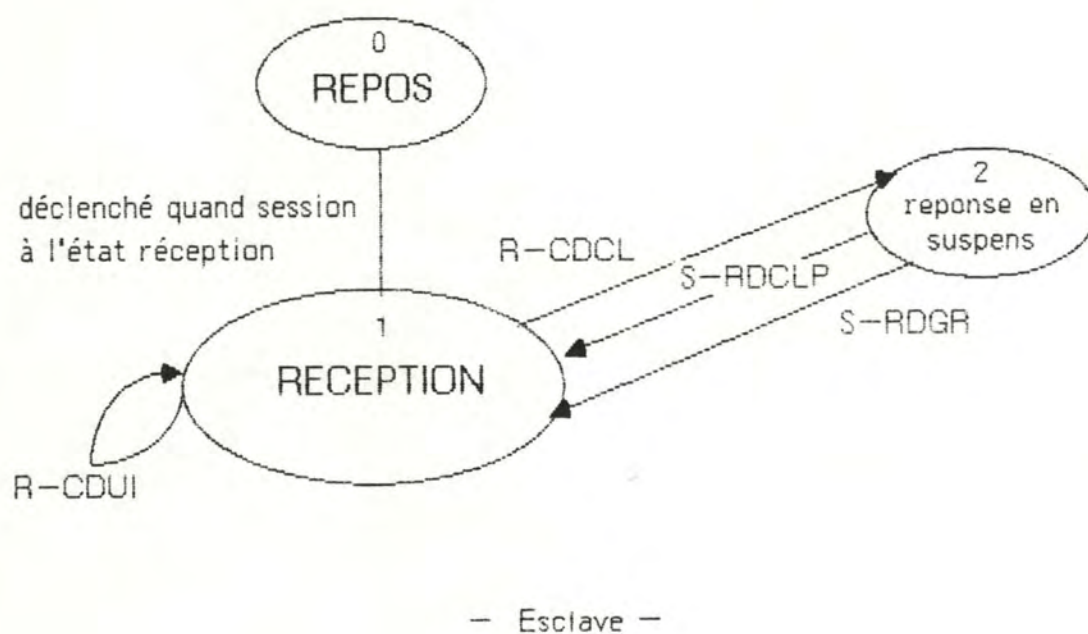
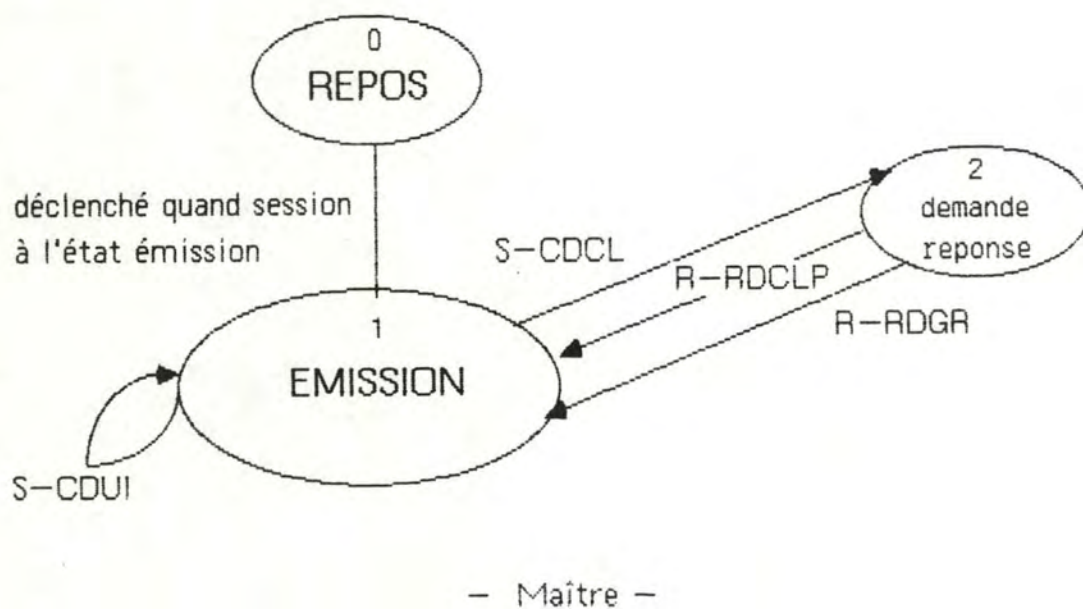


Figure 5.6: Graphes d'états et de transitions

5.3.2.2 Echange interactif

Le transfert sera réalisé par la commande CDUI.

Cette commande ne possède pas de paramètre.

CONCLUSION

L'objectif de ce mémoire était d'une part faire le point sur les principales interfaces physiques ainsi que sur les procédures de liaison de données (couches 1 et 2 de l'ISO) utilisées actuellement, d'autre part, d'établir un modèle pour l'échange interactif et l'échange de fichiers couvrant les couches 5 et 6 de l'ISO.

La première partie de ce mémoire pourra servir de référence à l'établissement des liaisons de données pour les équipements de traitement de données.

La deuxième partie pourra servir de base à la programmation future des deux applications envisagées : l'échange interactif et le transfert de fichiers.

En ce qui concerne la première partie, le chapitre intitulé "Interfaces physiques" reprend les principales interfaces utilisées pour la transmission de données par ligne téléphonique et par l'intermédiaire des réseaux publics.

Si l'on désire plus de renseignements sur les interfaces du RNIS, on pourra consulter le mémoire intitulé 'Etude de l'acheminement de l'information et de la signalisation pour accéder au RNIS', réalisé à cet institut par messieurs Georges et Sacré.

En ce qui concerne la deuxième partie, le modèle développé se base sur les couches supérieures de l'ISO et sur les couches 5 et 6 de l'application télétext.

La critique faite sur ces modèles a permis l'élaboration d'un système simple, réalisant les objectifs poursuivis.

Au cas où l'application nécessiterait de nouveaux services, ou bien qu'une étude complète de ces niveaux serait désirée, le modèle pourra être facilement élargi.

La segmentation des données, l'adressage et la préparation des documents n'ont pas été abordés dans le cadre de ce mémoire.

Ils font actuellement l'objet d'une étude réalisée par la firme qui nous a accueilli durant le stage. Cette étude comprend aussi les programmes d'application de transfert de fichiers et ceux de gestion et d'interrogation de bases de données.

BIBLIOGRAPHIE

CCITT

Projet avis S 62
AP VII-78-F

CCITT

Projet avis T 62
AP VIII-38-E

CCITT

X 25 - spécification
TELEPAC (Suisse)
1980

CCITT

X 28 - spécification
TELEPAC (Suisse)
1980

CCITT

X 21
Interface d'application générale entre
l'équipement terminal de traitement de
données (ETTD) et l'équipement de
terminaison du circuit de données (ETCD)
pour fonctionnement synchrone dans les
réseaux publics pour données

CCITT

X 21 bis
Utilisation, sur les réseaux publics pour
données, des équipements terminaux de
traitement de données (ETTD) destinés à
assurer l'interface des modems synchrones
de la série V

CCITT

X 20
Interface entre équipement terminal de
traitement de données (ETTD) et l'équipement
de terminaison du circuit de données (ETCD)
dans le cas des services avec transmission
arythmique sur réseaux publics pour données

CCITT

Avis V 25

Equipement d'appel et/ou réponse automatiques
sur le réseau téléphonique général avec
commutation, y compris la neutralisation des
suppresseurs d'écho lorsque l'appel est
établi entre postes à fonctionnement manuel

CEPT

Proposition for an interactive session protocol
based on S.62
Télétex Rapporteurs Group
Bern, 1-5 February 1982

Henrion P et Waltzing M

Les couches supérieures de l'ISO
Compatibilité avec différents types de réseau
Projet d'implémentation
Mémoire présenté en 1983 (FNDP)

Macchy et Guilbert

Téléinformatique
Dunod informatique
1979

Tanenbaum

Computer networks
Prentice Hall
1981